Maggio/Giugno 1989, Anno IV n. 3

La rivista per utenti di C-64/128 ed Amiga

COMMODORE GAZETTE

Informatica e ricerca:

L'AMIGA NELLE UNIVERSITÀ USA

Prove hardware:

- LE SCHEDE XT A2088 e AT A2286
- IL PC40-III
- LA STAMPANTE MPS 1230

Il mondo di GEOS:

GEOS 2.0 PER C-64 E C-128

Anteprima software:

CRIMETOWN DEPTHS, LA RISCOSSA DEL MADE IN ITALY

Programmare l'Amiga:

- IL LATTICE C 5.0
- L'EXTEND PER L'AMIGABASIC
- LE STRATEGIE DI TEMPORIZZAZIONE

Saloni USA:

DA CHICAGO IL COMDEX '89

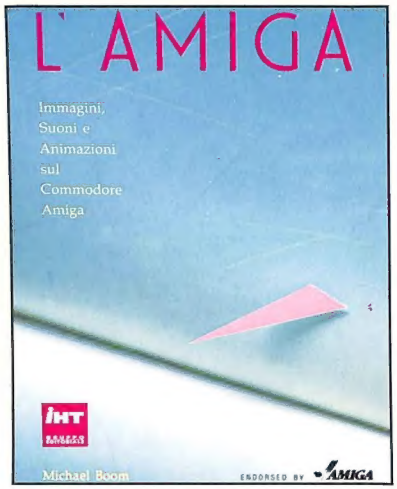
Listati:

MODO TESTO 80x50 per C-128

Sped. in abb. post. Gr. III/70 – Lire 8.000

NELLA FOTO L'UNIVERSITÀ AMERICANA DI BERKELEY

Un computer chiamato AMIGA



L'AMIGA

Come il Commodore Amiga ha cambiato il mondo dei computer, così il volume *L'Amiga* proietterà la vostra immaginazione lungo nuovi orizzonti di creatività. Preparatevi a ottenere stupefacenti risultati dal vostro computer: sofisticate immagini video, suoni e musica, sequenze animate da registrare su videocassetta, e molto di più. In questo volume troverete inoltre consigli utili per ottenere il meglio dall'Amiga Basic e da prodotti software eccezionali come Deluxe Paint, Deluxe Music e Deluxe Video.

416 pagine, oltre 100 illustrazioni, L. 60.000

IL MANUALE DELL'AMIGADOS

Questo volume è l'unica documentazione ufficiale realizzata dalla Commodore sul sistema operativo dei computer Amiga. Contiene tre libri (Il Manuale per l'utente, Il Manuale per il programmatore e Il Manuale di riferimento tecnico) che costituiscono nel loro complesso la guida più completa per ogni utente dell'Amiga, dal principiante che vuole usare i comandi dell'AmigaDOS (che non sono spiegati dall'opuscolo in dotazione al computer), fino al programmatore evoluto che troverà utili informazioni per programmare in C e in Assembly.

376 pagine, L. 60.000

I libri IHT sono disponibili nelle migliori librerie e computer shop. Per ordini diretti servirsi dell'apposito modulo pubblicato a pagina 127
IHT Gruppo Editoriale - Via Monte Napoleone, 9 - 20121 Milano - Tel. 02/794181-794122 - Fax 784021 - Telex 334261 IHT I
Distribuzione: Messaggerie Libri - Via Giulio Carcano, 32 - 20141 Milano - Tel. 02/8438141

PROGRAMMARE L'AMIGA VOL. II

Finalmente il volume *Programmare l'Amiga Vol. II* è disponibile! Le spedizioni alle librerie e ai privati che lo hanno ordinato direttamente avranno inizio il *15 settembre*.

L'edizione italiana è stata redatta apportando all'edizione in lingua inglese aggiornamenti e correzioni. Un lavoro che ha richiesto molti mesi, ma che oggi offre un prodotto decisamente migliore e più completo dell'edizione originale.

PER ORDINARE IL LIBRO COMPILARE E SPEDIRE IL TAGLIANDO
PUBBLICATO A PAGINA 127

AVVISO PER CHI HA GIÀ ORDINATO IL LIBRO

Tutti coloro che avessero già ordinato il libro direttamente presso la IHT sono pregati di leggere con attenzione quanto segue.
Chi ha ordinato il volume da non più di sei mesi lo riceverà direttamente a casa in contrassegno senza bisogno di nessuna ulteriore conferma.
Chi invece ha ordinato il volume da più di sei mesi, è pregato di voler spedire il tagliando di conferma sotto riportato.

Ritagliare e spedire a: IHT Gruppo Editoriale – Via Monte Napoleone, 9 – 20121 Milano

Nome e cognome: .....
Indirizzo: .....
Città: ..... Prov.: ..... C.a.p.: .....

☐ Desidero confermare il mio ordine per il libro *Programmare l'Amiga Vol. II* che ho già inoltrato nel periodo .....
Pagherò al postino la somma di Lire 70.000 + 6.000 (spese postali).

OPPURE:

☐ Desidero annullare il mio ordine per il libro *Programmare l'Amiga Vol. II* che ho già inoltrato nel periodo.....
Non riceverò nulla e quindi nulla vi è dovuto per l'ordine da me fatto.

Firma:.....

Articoli

FOTO PATRICIA LEEDS

42 **L'AMIGA NELLE UNIVERSITÀ AMERICANE**
L'Amiga diventerà il computer scientifico del futuro?

46 **L'ULTIMO NATO TRA I PC COMMODORE: IL PC40-III**
Un PC IBM compatibile di buon livello, con hard disk da 40 MB, grafica VGA e monitor bi-sync a fosfori bianchi

56 **LA RISCOSSA DEL SOFTWARE MADE IN ITALY**
Anteprima del gioco Crimetown Depths: Commodore Gazette intervista uno dei suoi creatori

61 **UNA COMMODORE PICCINA PICCINA AL COMDEX DI CHICAGO**
Aprile 1989: brutte notizie per la Commodore da un'importante mostra di sistemi professionali

64 **LE SCHEDE BRIDGEBOARD XT A2088 E AT A2286 PER L'AMIGA**
Un'accurata analisi della compatibilità MS-DOS per gli utenti dell'Amiga 2000

76 **I PACCHETTI GEOS 2.0 E GEOS 128 2.0**
Tutti i dettagli sulla nuova versione di GEOS, il sistema operativo per gli otto bit della Commodore

84 **LA MPS 1230 STUPISCE SOLO PER IL PREZZO: TROPPO ELEVATO**
Prove stampanti: grossa delusione per l'ultima proposta Commodore

89 **QUATTROMILA CARATTERI IN RGB PER IL C-128**
Un elaborato programma per raddoppiare la quantità di caratteri sul video del vostro C-128

94 **CORSO DI PROGRAMMAZIONE IN C PER L'AMIGA**
Quarta puntata: la versione 5.0 del compilatore C della Lattice

104 **L'ESTENSIONE EXTEND DELL'AMIGA BASIC**
Un pacchetto che amplia gli orizzonti dei programmatori Basic

110 **L'USO DEGLI INTERRUPT PER LE TEMPORIZZAZIONI DEI TASK**
Uno studio degli interrupt relativi ai CIA dell'Amiga, con un programma d'esempio

MAGGIO/GIUGNO 1989
Anno IV, N.3

Rubriche

5 NOTE EDITORIALI

6 LA POSTA DELLA GAZETTE

9 SOFTWARE GALLERY
C1-Text
Dark Side
Battletech
Dragon's Lair
Voyager
Ultima Trilogy
WordPerfect
Cosmic Pirate
Manhunter: New York
Time Scanner
Silkworm
Forgotten Worlds
Gunship
Star Trek
3D Pool

34 SOFTWARE HELPLINE
Leisure Suit Larry II

37 CLASSIFICA SOFTWARE

38 INPUT/OUTPUT

122 COMMODORE NEWS

124 CLASSIFIED

126 INDICE DEGLI INSERZIONISTI

127 SERVIZIO LETTORI

COMMODORE GAZETTE

La rivista per utenti di C-64/128 ed Amiga

COMMODORE GAZETTE

Una pubblicazione

IHT GRUPPO EDITORIALE

Direttore Responsabile
Massimiliano M. Lisa

Redazione
Luca Giachino (capo servizio redazione tecnica)
Mauro Gaffo (redattore)
Dario Greggio (redattore tecnico)
Nicolò Fontana Rava (servizi speciali)
Collaborazione Editoriale
Fabio Rossi, Marco Menichelli,
Filippo Moretti, Dario Zanon
Corrispondenti USA
William S. Freilich (sezione sviluppo)
Daniela D. Freilich (coordinamento generale)
Collaborazione Editoriale USA
Ervin Bobo, Eugene P. Mortimore,
Harriet Maybeck Tolly

Inviato speciale USA
Matthew Leeds

Segretaria di Redazione
Giovanna Varia

Impaginazione e Grafica
Antonio Gaviraghi (capo servizio)
Andrea De Michelis
Fotografia
A.&G.,
Patricia Leeds
Disegni
Marco Piazza, Giuseppe Festino, Lara Parravicini

Direzione, Redazione, Amministrazione
IHT Gruppo Editoriale S.r.l.
Via Monte Napoleone, 9 - 20121 Milano

Fotocomposizione
IHT Gruppo Editoriale S.r.l. - Reparto grafica
Fotolito
Colour Separation Trust S.r.l.
Via Melchiorre Gioia, 61 - 20124 Milano
Stampa
Nascor S.p.A.
Via Senigallia - 20161 Milano

Distribuzione per l'Italia
Messaggerie Periodici S.p.A.
V.le Famagosta, 75 - 20142 Milano
Tel. 02/8467545

Distribuzione per l'estero
A.I.E. S.p.A.
Via Gadames, 89 - 20151 Milano
Tel. 02/3012200

Pubblicità
IHT Gruppo Editoriale S.r.l.
Via Monte Napoleone, 9 - 20121 Milano
Ufficio Pubblicitario IHT
Italia ed Estero 02/794181-799492-792612-794122
Fax 02/784021 - Telex 334261 IHT I

Servizio Abbonamenti

IHT Gruppo Editoriale
Servizio Abbonati
Via Monte Napoleone, 9 - 20121 Milano

SEGRETERIA ABBONAMENTI
Linea per registrazione abbonamenti
02/794181-799492-792612-794122

Commodore Gazette
Costo abbonamenti

Italia
12 numeri L. 96.000
24 numeri L. 192.000
36 numeri L. 288.000
Estero:
Europa L. 150.000 (10 numeri)
Americhe, Asia... L. 200.000 (10 numeri)

I versamenti devono essere indirizzati a:
IHT Gruppo Editoriale S.r.l.
Via Monte Napoleone, 9 - 20121 Milano
mediante emissione di assegno bancario o vaglia postale

Arretrati
Ogni numero arretrato: L. 16.000 (sped. compresa)

Autorizzazione alla Pubblicazione
Tribunale di Milano n. 623 del 21/12/85
Periodico Mensile - Sped. in abb. post. gr. III/70
ISSN: 0394-6991

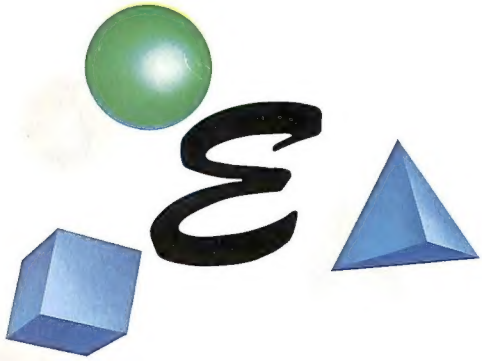
La IHT Gruppo Editoriale è iscritta nel Registro
Nazionale della Stampa al n. 2148 vol. 22
foglio 377 in data 5/6/1987

Commodore Gazette è una pubblicazione
IHT Gruppo Editoriale
Copyright © 1989 by IHT Gruppo Editoriale S.r.l.
Tutti i diritti riservati.
Nessuna parte della rivista può essere in alcun modo riprodotta senza autorizzazione scritta della IHT Gruppo Editoriale. Manoscritti e foto originali, anche se non pubblicati, non si restituiscono.
I contributi editoriali (di qualunque forma), anche se non utilizzati, non si restituiscono.
Non si assume alcuna responsabilità per eventuali errori od omissioni di qualsiasi tipo.

Commodore Gazette è un periodico indipendente non connesso in alcun modo con la Commodore Business Machines Inc. né con la Commodore Italiana S.p.A. PET, CBM, Vic-20, C-64, C-128, Amiga... sono marchi protetti della Commodore Business Machines. Nomi e marchi protetti sono citati senza indicare i relativi brevetti. Macintosh è un marchio della Apple, IBM PC AT, XT... sono marchi protetti dalla International Business Machines.

Associato alla U.S.P.I. (Unione Stampa Periodica Italiana)

NOTE EDITORIALI



L'impegno da parte della redazione di *Commodore Gazette* per offrire ai lettori una rivista sempre migliore è in continua crescita. Le recenti novità come la maggiore presenza di recensioni hardware e software e la serie di articoli che riguardano la programmazione dell'Amiga non sono le uniche. Da oggi infatti, oltre all'aspetto del contenuto editoriale del periodico, particolare attenzione dovrà essere prestata per quel che riguarda l'organizzazione della presenza in redazione dell'hardware Commodore.

Fino a qualche tempo fa, seppure con molte difficoltà quasi tutto l'hardware presente in redazione era materiale della Commodore Italiana inviato in visione. L'abitudine della Casa editrice era poi quello di restituire l'hardware al momento della sua uscita dal listino ufficiale. Questo per l'ovvio motivo che una rivista dedicata interamente ai prodotti Commodore non parla una sola volta di un prodotto. Una rivista come *Commodore Gazette* dopo aver provato un Amiga 500 non lo regala ai suoi collaboratori. Tiene invece la macchina in redazione pronta per verificare se per esempio il tal programma funziona correttamente, per effettuare la prova delle periferiche nate per affiancarsi al computer, per valutare eventuali problemi d'incompatibilità con i modelli successivi, per rispondere alle domande dei lettori...

Ma le cose sono cambiate. La Commodore notoriamente anche a livello internazionale ha sempre compiuto delle scelte di promozione quantomeno particolari: fa scuola l'impiego di un Macintosh al posto del previsto Amiga in alcune sequenze del film *Star Trek IV*, perché nonostante la presenza del computer nelle scene fosse del tutto gratuita, la disponibilità della Casa madre statunitense a inviare una macchina alla Paramount era molto tiepida. E allora? Allora si è pensato bene di allinearsi con la generale tendenza poco simpatica nei confronti di chi può contribuire alla maggior diffusione dei prodotti Commodore. D'ora innanzi quindi oltre al fatto che il supporto economico a una rivista come *Commodore Gazette* è pressoché inesistente (la Commodore fa pubblicità sulla rivista in media una volta all'anno) si è pensato bene d'imporre limitazioni nell'invio di hardware.

Per prima cosa un *aut aut*: o si restituisce tutto il materiale attualmente in redazione oppure lo si acquista con uno scambio. L'hardware in cambio di quelle poche pagine di pubblicità che la CBM acquista ogni tanto (il che è come dire: la Commodore non darà mai più una lira a *Commodore Gazette*, neanche una volta all'anno!). Poi la novità: il materiale che verrà inviato in visione (che sarà accuratamente centellinato) dovrà essere restituito entro tre mesi.

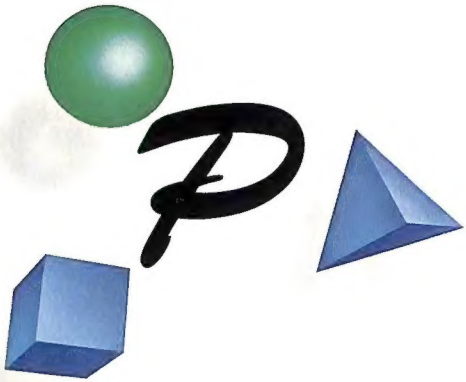
Da questo nasce l'esigenza di un particolare tipo di organizzazione. Visto il rapido turnover delle macchine inviate, la loro presenza in redazione va attentamente pianificata, prendendo in esame la possibilità di farsi inviare nuovamente lo stesso hardware, di far prorogare le scadenze ufficiali, di richiedere l'hardware ai negozi...

Vi aspettavate che la Commodore Italiana desse maggior supporto alla vostra rivista preferita? Sbagliavate. Dietro ogni numero c'è un grandissimo lavoro che quasi mai viene agevolato dalla Commodore Italiana. Anzi...

ML

LA POSTA DELLA GAZETTE

LA VOCE DEI NOSTRI LETTORI



IN DIFESA DELLA SIMULMONDO

Nel leggere le recensioni dei giochi non mi trovo d'accordo su quanto asserisce il signor F.R. riguardo a *Italy '90 Soccer*, anzi, leggendo la recensione di *Zany Golf* mi viene un forte sospetto: che il succitato F.R. abbia qualcosa contro Simulmondo?

Infatti nella sua recensione distrugge in pratica *Italy Soccer*, ma leggendo le sue righe mi viene il dubbio che non sia stato in grado di giocarlo e capirlo appieno. In primis egli asserisce di non aver notato variazioni con l'uso dell'opzione "strategie" quando in realtà è fondamentale. Provi a fare alcune partite cambiando tattiche e vedrà che qualcosa cambia. In secondo luogo liquida con estrema facilità il fatto che tramite dei cursori RGB si possano variare le divise delle squadre (notare che si possono addirittura far giocare squadre con divise di club come la Juve, l'Inter, il Milan), non mi sembra che in circolazione per l'Amiga o per altri computer vi siano simulazioni che offrono le stesse caratteristiche. Poi mi sembra addirittura ridicolo ciò che si dice a proposito del replay. L'indietro veloce che cita sempre il fantomatico signor F.R., in realtà, è un'opzione che permette di posizionare la scena al punto che si vuole e quindi da lì rivedere tutta la scena al rallentatore, cosa che mi sembra non faccia *Microprose Soccer*, che pure reputo il secondo miglior emulatore di calcio per Amiga. In conclusione, e premettendo che non sono pagato da Simulmondo per difendere i suoi games, vorrei criticarvi in quanto vi lamentate che in Italia non esistono software house in grado di proporre programmi di livello internazionale, e poi quando qualcuno propone qualcosa di valido viene criticato senza pietà.

Da notare che nel panorama dei games di calcio per l'Amiga, se togliamo *Microprose Soccer* che tra l'altro non è affatto eccezionale come si vuole far credere, i vari *P.B. International Soccer*, *Amiga Soccer*, *Soccer King*, fanno veramente pena.

Gianni Santi
Cagli (PS)

Sono molto dispiaciuto di non aver ottenuto il suo consenso con la mia recensione di Italy '90 Soccer*: dopo aver rigiocato per parecchie ore il programma in questione ho comunque appurato che alcune delle sue critiche sono assolutamente corrette. Temo di aver inserito alcune inesattezze nel mio articolo, e colgo l'occasione per scusarmi di ciò con lei e con tutti gli altri lettori. Passo quindi immediatamente a esaminare le sue affermazioni.*

1) Il menu RGB permette in effetti di ridefinire i colori delle maglie dei giocatori secondo strisce verticali, con gli effetti da lei indicati. Ancor oggi mi chiedo come possa essermi sfuggita una tale particolarità, e riconosco il mio errore.

2) Per quanto riguarda il peggioramento progressivo degli effetti sonori (quella frase mi è stata criticata da altre fonti), devo attribuire lo strano evento alla macchina sulla

Indirizzate tutta la corrispondenza per la rivista a:

COMMODORE GAZETTE
La posta della Gazette
Via Monte Napoleone, 9
20121 Milano

Preghiamo i lettori di essere concisi e concreti, per darci modo di rispondere al più grande numero possibile di lettere. La redazione si riserva comunque il diritto di sintetizzare le lettere troppo lunghe.

quale ho eseguito i test del gioco. Provando il programma su di un altro computer ho potuto constatare l'assenza del fenomeno, evidentemente causato da qualche bizzarra idiosincrasia del mio Amiga 1000. È comunque da notare che fino a oggi non si erano mai verificati problemi analoghi con le centinaia di programmi da me caricati sulla stessa macchina.

3) Il replay. Anche in questo caso ho dovuto testare il programma su di un secondo computer per poter utilizzare l'opzione. A questo punto devo però porle una domanda: secondo lei è da giudicare positivamente un programma che funziona in maniera del tutto diversa a seconda del computer sul quale viene caricato? Badi che non sto parlando di diversi modelli (Amiga 1000, 2000 o 500) ma di due diversi Amiga 1000, e il perfetto funzionamento del primo computer utilizzato per i test è stato accertato in più modi.

Temo inoltre di dovere contestare il giudizio che lei dà sulle strategie: posso assicurarle che ho giocato a Italy '90 Soccer *per molto tempo e utilizzando diverse opzioni, ma non sono riuscito a riscontrare differenze sostanziali con l'impiego di diverse strategie selezionate secondo i dettami del manuale.*

Prima di chiudere vorrei inoltre puntualizzare un'ultima cosa: ogni gioco che mi viene affidato per essere recensito viene da me provato e riprovato, giocato quasi sempre in tutte le sue fasi e comunque esaminato in modo approfondito. I miei giudizi tengono sovente conto del parere di terzi (colleghi, amici e persino parenti), e anche per questo motivo sono esenti da partigianerie, simpatie e antipatie di qualsivoglia genere. Così come lei non è stato pagato dalla Simulmondo per scrivere questa lettera, io non vengo pagato da nessuno per distruggere o meno determinati prodotti. Gli unici soldi che ricevo sono (indirettamente) i suoi e quelli dei lettori che acquistano Commodore Gazette, *per un servizio che cerco di svolgere con la massima professionalità possibile, documentandomi a lungo prima di scrivere anche una sola riga dei miei articoli. Favorire un prodotto a parer mio mediocre di una software house italiana, in un panorama di programmi spesso di alta qualità provenienti dall'estero, significherebbe dimenticare l'obiettività e fornire un giudizio falsato che scontenterebbe molti lettori.*

Detto questo, sarò ben lieto di attribuire i massimi voti ai prossimi prodotti Simulmondo, se saranno meritati.

Fabio Rossi

CHI SE LA SENTE DI RISPONDERE?

Nel lontano giugno 1988 sono riuscito a farmi regalare un Amiga 500 completo di monitor, sicuro di aver acquistato un ottimo computer. Dopo aver passato luglio e agosto al mare, a settembre ho iniziato a divertirmi con il computer: tutto è filato liscio sino a quando il 12/11/88 l'Amiga non si è guastato. L'ho portato dal negoziante e insieme abbiamo deciso di spedirlo al più vicino Centro Assistenza (Napoli)... dopo tre mesi nessuna notizia, perché il negoziante aspettava che lo spedissero dal Centro e il Centro aspettava che venisse a prenderlo il negoziante. Dopo circa quattro viaggi a Napoli me lo sono ripreso tirando un sospiro di sollievo. Ma entro una settimana (usato due volte) si è rotto di nuovo (siamo a fine gennaio). Altri due mesi a Napoli e quando l'ho riportato a casa non funzionava: con il negoziante che me l'aveva venduto abbiamo constatato che mancavano alcuni pezzi. Siamo al 20 febbraio. Dopo numerosi viavai mi sento dire che il mio computer fa parte di una serie che presenta un guasto ricorrente. Dopo un altro mese riparano gli altri ma il mio si guasta di nuovo davanti ai miei occhi e a quelli del negoziante: mi viene detto che se aspetto un altro po' sarebbero arrivati altri componenti e l'avrebbero riparato... ora, dopo tre mesi, mi sento dire che i componenti che servono non li hanno neppure a Milano e che se ne riparlerà tra uno o due mesi...

Ormai mi sono rassegnato (almeno ho perso con onore), ma mi piacerebbe avere qualche risposta, sapere cosa mi aspetta. O forse ho suscitato la pietà di qualcuno e un giorno tornando a casa vedrò sulla mia scrivania un bell'Amiga (e vissero felici e contenti)...

Davide Landi
Baronissi (SA)

IMMAGINI, SUONI E ANIMAZIONI

Premettendo che sono un possessore di un Commodore 64C, di un Commodore Amiga 500 e di un Atari 800 ST, vi pongo i seguenti quesiti.

1) Con le macchine in mio possesso è possibile creare grafica servendosi di apparecchiature come videolettori, penne ottiche, tavolette grafiche e così via? Posso poi animarle e manipolarle a mio piacimento?

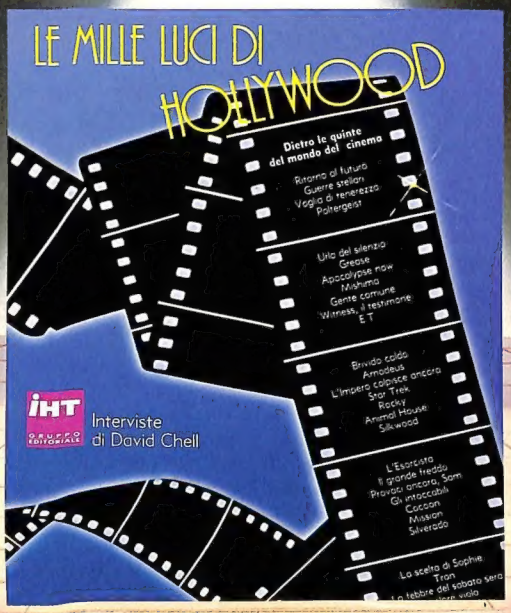
2) Per creare la grafica di sfondo e per ridefinire la pagina grafica che programmi devo usare?

3) Con la musica come devo fare, cosa devo usare?

Adriano Giovannini
Sesto Fiorentino (FI)

Proprio per venire incontro a esigenze come le sue la Casa editrice di Commodore Gazette *ha pubblicato il libro* L'Amiga: Immagini, Suoni e Animazioni sul Commodore Amiga. *Ci permettiamo di consigliarglielo.* ■

REGALA IL CINEMA



Un'inedita e affascinante raccolta d'interviste ai più famosi "creatori di cinema" degli anni '80. Questa volta non sono al centro dell'attenzione attori e registi, ma i tecnici. Tecnici premiati con l'Oscar, esperti in campi così particolari che molti di noi non ne sospettano neppure l'esistenza. Per curiosare dietro le quinte del mondo di celluloide. Per scoprire come si è trasformata la Hollywood degli anni '80. Per avere notizie sul futuro della computergrafica e sulle ultime novità nel campo degli effetti speciali. Ma anche per imparare ad amare il cinema e conoscerlo meglio.

I libri IHT sono disponibili nelle migliori librerie. Per ordini diretti servirsi dell'apposito modulo pubblicato a pagina 127
IHT Gruppo Editoriale - Via Monte Napoleone, 9 - 20121 Milano - Tel. 02/794181-794122 - Fax 784021 - Telex 334261 IHT I
Distribuzione: Messaggerie Libri - Via Giulio Carcano, 32 - 20141 Milano - Tel. 02/8438141

SOFTWARE GALLERY

UNA GUIDA PER ORIENTARSI NEL MONDO DEL SOFTWARE

C1-TEXT

Un word processor italiano di elevato livello a un prezzo estremamente competitivo

Computer: Amiga
Supporto: Disco
Prezzo: L. 69.000
Produzione: Cloanto
Disponibile presso: Leader (Via Mazzini 15, 21020 Casciago – 0332/212255)

Eravamo al corrente della sua esistenza, ma non sapevamo quando avremmo potuto finalmente utilizzarlo. Poi è giunto in redazione e, anche se non spiccava troppo tra le coloratissime copertine dei pacchetti software pronti per la prova, non abbiamo saputo resistere alla curiosità e gli abbiamo dato la precedenza su tutti gli altri.

Stiamo parlando del word processor Italiano (maiuscolo, perché è un vero portabandiera nazionale) scritto e realizzato dalla Cloanto di Udine. Il prodotto è stato ultimato nei primi mesi di quest'anno ed è stato presentato ai visitatori di diverse mostre di computer. Il manuale che lo accompagna lo descrive come il primo word processor realmente pensato per gli utenti di qualsiasi nazione. È stata riposta la massima cura per facilitare le operazioni più ripetitive, per rendere quanto più semplice possibile l'interfaccia con la macchina e per consentire tutte le operazioni classiche dei word processor.

SCHEDA CRITICA

INSUFFICIENTE
Un pessimo prodotto che non merita nessuna considerazione.

MEDIOCRE
Il programma ha alcuni difetti di fondo anche se nel complesso raggiunge quasi la sufficienza

DISCRETO
Un prodotto accettabile ma non aspettatevi grandissime emozioni

BUONO
Raccomandato vivamente tra i migliori programmi della sua categoria

OTTIMO
Eccezionale! Fino ad oggi non si era mai visto niente del genere

L'involucro del pacchetto ci rende subito orgogliosi della nostra nazionalità, ricoperto com'è di fasce tricolori, e questo orgoglio aumenta leggendo la scritta "Il primo word processor italiano" al centro della custodia. È un orgoglio che nasce dalla constatazione che anche noi sappiamo realizzare prodotti unici e superiori, dopo anni di senso d'inferiorità nei confronti degli stranieri (specialmente degli americani) per tutto quello che riguardava la produzione di software.

All'interno della confezione troviamo il manuale di circa 120 pagine, i due dischi del programma e alcune note sul copyright e sulla possibilità di essere registrati presso la Cloanto come utenti del suo software. Il tutto è ricoperto da un doppio strato di spugna.

Il programma parte automaticamente dopo il boot dal disco 1. Ovviamente, trattandosi di un editor scritto e pensato per l'Amiga, tutte le funzioni sono accessibili tramite menu a tendina, e in buona parte anche tramite i tasti Amiga in combinazione con qualche altro tasto o con i tasti funzione (il cui significato è visualizzato in maniera un po' oscura dalle icone nella fascia bassa dello schermo, che si possono rimuove-

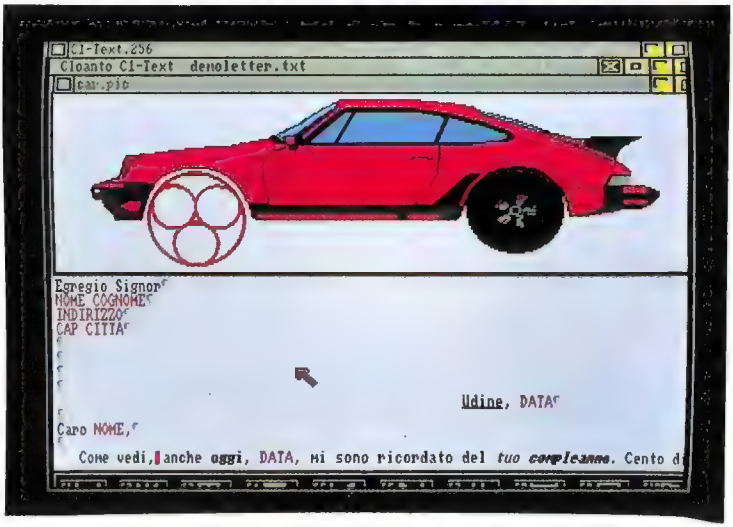
re in qualunque momento). Prima di leggere il manuale, digitiamo due o tre parole per provare il feeling del primo impatto, e subito osserviamo con piacere che il programma ha modificato per noi l'iniziale del paragrafo rendendola maiuscola e ha inoltre trasformato un "pò" nel corretto "po'": constatare l'efficacia del correttore di errori ci fa un piacere immenso.

Forti di quest'impressione favorevole, passiamo a esaminare il manuale. La qualità generale è piuttosto curata ed è stata adottata una nomenclatura standard per tutti i termini italiani e inglesi del gergo "computerese", il che evita equivoci o oscurità. A volte si verifica qualche accostamento che fa sorridere, come nel caso dei "bottoni testo" (gadget stringa), ma le scelte adottate ci sembrano generalmente valide. Il manuale non è un tutorial, ma inizia con una parte d'introduzione al computer in generale e all'Amiga in particolare, proseguendo poi con l'esame di tutte le possibilità offerte dal pacchetto nell'ordine presentato dai menu a tendina. Riscontriamo purtroppo la mancanza di una sezione che descriva il contenuto dei dischi, mentre ci si dilunga per parecchie pagine nella spiegazione di problemi inerenti la licenza d'uso del programma.

Iniziamo la descrizione del programma con il primo menu a sinistra, Generale, da cui possiamo selezionare le opzioni relative al caricamento e alla memorizzazione dei file di testo, e alla gestione di tutte le operazioni precedenti o successive alla scrittura vera e propria. Le operazioni che richiedono la selezione di un file da disco fanno uso di un ampio requester nel quale vengono presentati tutti i file esistenti sul dispositivo scelto (dispositivo che può essere anche il RAD:, il disco virtuale resistente al reset nato con la versione 1.3 del *Workbench*). Nel caso di un file da salvare viene anche mostrato un ventaglio di possibili formati di memorizzazione (non è purtroppo possibile salvare uno specifico blocco di testo).

La varietà di formati disponibili dimostra l'attenzione prestata durante la realizzazione di *C1-Text* al problema della mancanza di standard nella codifica dei file. Il pacchetto è infatti in grado di riconoscere una nutrita serie di modi, sia per il formato dei file sia per ciò che concerne la codifica dei caratteri al loro interno. Vengono riconosciuti i modi IFF, ASCII standard e ANSI X3.64, e un particolare formato compresso che consente la riduzione delle dimensioni del file fino al 50 per cento. È presente anche un formato crittografato, basato su un doppio algoritmo di codifica. Inoltre è possibile ridirigere in un file l'output che normalmente andrebbe inviato alla stampante mediante il menu di stampa, la qual cosa permette di sopperire alla mancanza di una preview a video del risultato finale.

Questo word processor, infatti, non è un vero WYSIWYG (What You See is What You Get), in quanto vengono mostrati a video solo la posizione dei margini e gli stili usati per i caratteri (peraltro tratti da una sola fonte-carattere) ma non le formattazioni particolari, il numero di pagina, le testatine e così via. In compenso la gestione del cursore è piuttosto agevole: il mouse può essere utilizzato per portare il cursore in qualsiasi punto dello schermo (anche a destra di un ritorno carrello) e per far scorrere la parte visibile del testo, oltre che per selezionare i blocchi. Sono poi disponibili opzioni dei menu sia per spostarsi a una certa riga, sia per muoversi di pagina in pagina e di schermata in schermata. Una soluzione che abbiamo trovato poco immediata è la sequenza di tasti funzione associata alle operazioni di spostamento del cursore: sequenze come F5-F5 o F5-F9 per spostarsi all'inizio del file o per andare alla pagina precedente sono un po' meno facili da ricordare delle consuete CTRL-HOME



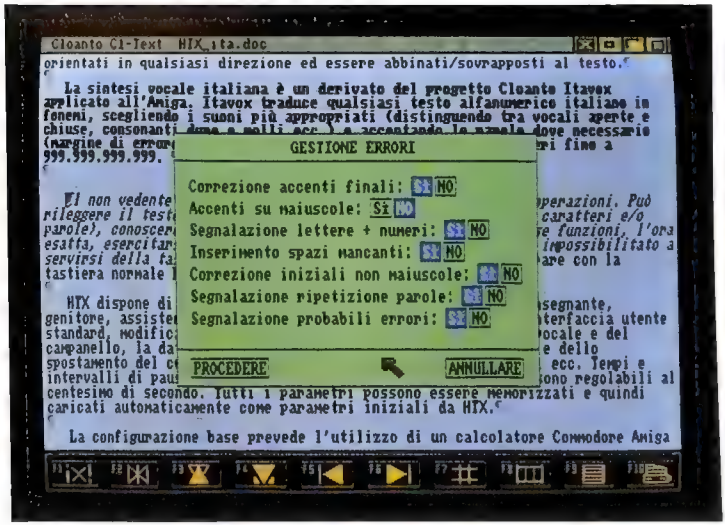
o PgUp usate da quasi tutti i word processor evoluti. Inoltre non esistono comandi per muoversi alla parola o alla frase precedente o successiva.

Nel menu Testo troviamo le già citate operazioni relative allo spostamento del cursore e quelle per il cut & paste e per la ricerca e la sostituzione delle stringhe. Mentre il cut & paste funziona nel modo classico, particolare cura è stata prestata alla routine di ricerca, basata su una variante della "ricerca di espressioni regolari" o *grep* dell'ambiente Unix. Vengono infatti riconosciuti diversi caratteri jolly, si può decidere se rendere significativa la distinzione tra maiuscole e minuscole e ogni volta che la stringa viene individuata sono possibili svariate scelte, tra cui l'insolito conteggio del numero di ricorrenze della stringa nel testo.

Sempre da questo menu è possibile lavorare su un blocco di caratteri alterandone lo stile o eliminando eventuali corsivi, neretti e così via. È anche possibile effettuare l'ordinamento alfabetico delle righe, un'altra di quelle caratteristiche inusuali ma apprezzabilissime in un word processor (anche perché permette di escludere, dal punto di vista della chiave di ordinamento, le prime lettere di ogni riga).

Il menu di stampa serve ad attivare la stampa di un file o di una sua parte, oppure a ottenere la stampa di un'immagine IFF presente in memoria o caricabile da disco. A proposito delle immagini, bisogna dire che non è possibile fondere testo e grafica, né tantomeno effettuare operazione tipiche del desktop publishing come la sagomatura del testo intorno a una figura. Le immagini possono soltanto essere visualizzate e spostate all'interno di una finestra loro dedicata, ma non si possono alterarne i colori o effettuarne il ridimensionamento; alla fine possono essere stampate, eventualmente sospendendo la stampa del testo e inserendole nella posizione desiderata all'interno del documento.

Per i parametri di stampa occorre rivolgersi a una sotto-voce del menu successivo, quello dedicato ai parametri globali. Ricordiamo ancora che la stampa del testo può essere ridiretta in un file per sopperire alla non completa corrispondenza tra immagine a video e immagine stampata. Si possono selezionare da questo menu diverse funzioni, come la stampa delle sole pagine pari o delle sole dispari, il numero di copie, l'uso dei caratteri di default (selezionati da Preferences) o di altri scelti tra quelli offerti dalla stampante, e così via. Inoltre vi sono alcune caratteristiche volte principalmente a migliorare la praticità, come la visualizzazione di un requester per richiedere la sostituzione del foglio singolo o l'abilitazione/disabilitazione dell'avvisatore acustico della stampante.



Dal menu Parametri è anche possibile definire le caratteristiche di stampa del documento: si può attivare la numerazione (in numeri arabi o romani, e in diverse posizioni della pagina) e la presenza di un'intestazione comune a tutte le pagine. Si possono impostare spazi aggiuntivi su entrambi i margini e si può decidere quanto spazio lasciare nella parte alta e in quella bassa di ogni pagina. Purtroppo non si possono avere note a piè di pagina o parole ripetute al fondo di ogni pagina. Si possono impostare da questo menu anche i margini del testo e il tipo di allineamento: con sbandieramento a destra o a sinistra, centrato o giustificato a pacchetto. Come abbiamo detto, a video non vengono visualizzati tutti questi aspetti dello stampato, ma è possibile sapere quale sarà l'allineamento di un dato paragrafo mediante l'indicazione fornita da una lettera che compare in alto a destra.

Il menu Parametri porta poi anche quelle personalizzazioni dell'interfaccia utente che avevamo citato come uno dei punti di forza del pacchetto. Si possono

alterare formato della data, volume dell'audio (descritto come stereofonico ma limitato a semplici rumori che si odono quando il sistema ci segnala qualcosa), i colori dello schermo di *C1-Text*, nonché la lingua dei messaggi del sistema (nelle versioni abilitate per più di una lingua che sono distribuite nei Paesi come la Svizzera), quella usata per determinare le regole ortografiche del testo in fase di scrittura e la mappa di caratteri della tastiera.

A questo proposito torniamo a parlare della notevole funzione di correzione degli errori che il programma è in grado di svolgere sui nostri testi. Nel caso della lingua italiana, vengono riconosciuti, in base a un vocabolario interno a cui l'utente non può accedere e che quindi non può essere ampliato, gli esatti accenti di moltissime parole, l'esatta divisione in sillabe delle parole per andare a capo correttamente e alcuni apostrofi (come "un po'"). Nelle versioni create per le altre nazioni sono previste le regole proprie di quelle lingue, come la dieresi per il tedesco. Tra gli altri automatismi, ovviamente tutti disattivabili permanentemente ricorrendo all'apposita opzione, o temporaneamente con il tasto HELP, citiamo la maiuscola automatica dopo un punto, la segnalazione di eventuali ripetizioni e il controllo dell'esattezza delle parole mentre vengono scritte.

Ci restano ancora da esaminare la funzione di mail merge e la crittografia.

La prima è abbastanza consueta, in quanto fa affidamento su un file di dati composto mediante record (la cui struttura è personalizzabile) e su certe parole chiave scritte in rosso nel testo. In fase di stampa, per ogni copia del documento le parti in rosso vengono sostituite con i relativi campi del file di dati; pur non essendo possibile la sostituzione condizionale dei campi, si può fare in modo che *C1-Text* ne richieda di volta in volta il contenuto.

La funzione di crittografia è invece inconsueta: si tratta di una funzione alquanto sofisticata che facendo uso di una chiave di codifica lunga fino a 50 caratteri e fornita dall'utente, altera il testo originale facendolo diventare irriconoscibile.

Dopo più di mezza giornata di prove, siamo ormai giunti alla fase finale e ci aggiriamo tra i menu per vedere se ci è sfuggito qualche particolare. Ne approfittiamo per attivare l'opzione Statistiche per vedere alcuni dati relativi al testo appena scritto. Tutto finora ci è sembrato molto bello e certe idee ci hanno addirittura entusiasmato, ma voltandoci un attimo verso l'altra scrivania, sulla quale è appoggiato *WordPerfect*, l'altro word processor in recensito in questo stesso numero, alcuni dubbi iniziano a farsi strada nel luminoso panorama delineato finora.

È possibile che una confezione grossa il doppio, con un manuale d'uso al cui confronto quello del programma italiano sembra un volantino pubblicitario, non abbia niente di più del sin qui completissimo *C1-Text*? Basta poco per renderci conto che certe cose non hanno nemmeno sfiorato la mente dei programmatori della Cloanto: per esempio le macro di comandi, o la possibilità di aprire 32 file in altrettante finestre, lo spell-checker con tanto di dizionario dei sinonimi, le funzioni matematiche o la suddivisione del testo in colonne. Ma va rilevato che la differenza di prezzo tra i due prodotti è davvero abissale, e al prezzo proposto *C1-Text* risulta essere un programma eccellente. Ciò non toglie che la pretesa di assoluta completezza ripetuta più volte nelle prime pagine del manuale viene a perdere un po' di credibilità in seguito alle ultime considerazioni.

Questo non sminuisce le qualità del programma, che è senza dubbio un ottimo prodotto per chi scrive, anche se si dimostra più legato alla scrittura fine a stessa che al word processing in generale. Questo si nota per esempio nell'assenza della possibilità di eseguire comandi AmigaDOS dall'interno dell'editor, o nel fatto che nessun prompt ci avvisa se stiamo inavvertitamente sovrascrivendo un file già presente. D'altra parte le correzioni automatiche durante la battitura sono comodissime, e bug non ne abbiamo trovati.

Rimane però un dubbio, a questo punto. Ci siamo infatti resi conto, in una fase successiva alla prova vera e propria, che alcuni particolari non quadrano. Per esempio le foto stampate sulla confezione di *C1-Text* non rispecchiano le reali capacità del programma, in quanto mostrano una sessione di lavoro nella quale vengono aperte più finestre di controllo, il che è impossibile. Analogamente, l'affermazione che il manuale è stato stampato interamente con questo word processor è fuorviante perché, tra l'altro, non è possibile ottenere testatine diverse nelle pagine pari e in quelle dispari. Inoltre, e questo ci ha stupiti moltissimo, nel manuale si afferma che il disco sistema contiene alcune utility Unix offerte gratuitamente dai programmatori, mentre è facile scoprire che questi nuovi comandi sono stati realizzati semplicemente cambiando nome ai relativi comandi dell'AmigaDOS.

Tutto ciò non sminuisce naturalmente il valore del pacchetto, ma dimostra scelte quantomeno poco felici da parte della Cloanto. Si tratta di soluzioni propagandistiche che dispiace dover rilevare; soprattutto considerando che *C1-Text* è uno dei primi programmi italiani di livello veramente internazionale.

D.G.

DARK SIDE

Tecnica Freescape e grande giocabilità, ma sembra una copia carbone di Driller...

Computer: Amiga
Supporto: Disco
Prezzo: L. 59.000
Produzione: Incentive
Distribuzione: Leader (Via Mazzini 15, 21020 Casciago – 0332/212255)

Evath è un pianeta che si trova a miliardi di chilometri dalla Terra che è stato scelto per diventare la nuova patria della razza umana. Per gli esseri umani Evath rappresenta infatti l'unica possibilità di non estinguersi insieme con il pianeta natale e ne va quindi garantita a ogni costo la sicurezza.

Per questo motivo siete stati scelti per una pericolosa missione su Tricuspid, una delle lune del pianeta, con il compito d'impedire ai banditi di Ketar di distruggere Evath con un potentissimo raggio a particelle.

A chi ha sentito parlare di *Driller*, un gioco prodotto dalla stessa casa, quest'ambientazione non dovrebbe apparire nuova, visto che il gioco si svolge sullo stesso pianeta a qualche anno di distanza. Come il suo predecessore, anche questo programma si avvale della tecnica Freescape, che permette al computer di rappresentare in maniera molto realistica un mondo costituito da solidi tridimensionali "pieni", fra i quali il giocatore si può muovere in tutte le direzioni, riuscendo a osservare gli elementi del paesaggio da ogni angolazione e anche entrando al loro interno. In realtà, la grafica risulta piuttosto sacrificata rispetto alla media dei programmi per l'Amiga, ma la complessità dello scenario giustifica in parte questa carenza.

Non giustifica però la scarsa fantasia nell'elaborazione di nuove situazioni di gioco: chi già possiede *Driller* non troverà infatti in *Dark Side* significative differenze dal punto di vista dell'azione, della strategia, della grafica e, in generale, dello svolgimento del gioco. In realtà, l'unica cosa che in parte lo differenzia dal predecessore è l'obiettivo (e ci mancherebbe altro!) ma si tratta di una differenza più teorica che sostanziale. Mentre infatti lo scopo di *Driller* consisteva nello scavare bocche di uscita per le sacche di gas che rendevano instabile la luna Mitral, l'obiettivo di *Dark Side* è di abbattere il più rapidamente possibile alcuni collettori di energia (ECD) che alimentano la potenza distruttrice del raggio Ketar. Com'è ovvio, la missione va compiuta il più rapidamente possibile per evitare il disastro... per il resto, premere un tasto per attivare una trivella o per sparare con un laser non è molto diverso.

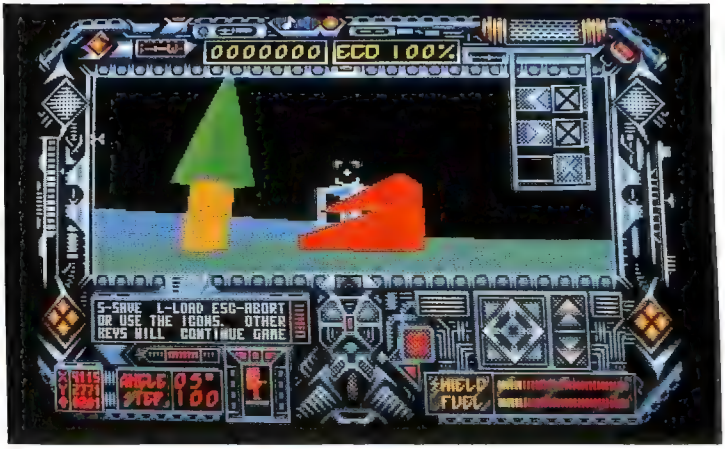
Anche il campo di gioco è praticamente invariato, per quanto ci si trovi sulla luna Tricuspid, anziché su Mitral.

Per quanto riguarda il mezzo affidato in dotazione al giocatore, ci sono anche in questo caso poche differenze; pur trattandosi infatti del trasportatore jet individuale Jetpac, il "veicolo" presenta caratteristiche di mobilità e manovrabilità praticamente identiche al carro di *Driller*.

Restano confermate, comunque, anche le buone caratteristiche di giocabilità e di strategia che contraddistinguevano il predecessore di *Dark Side*, che, unite al buon effetto tridimensionale del sistema Freescape, fanno di questo programma sicuramente un buon prodotto.

L'unico suggerimento che possiamo dare è quindi il seguente: i possessori di *Driller* evitino l'acquisto di quello che sarebbe in tutti i sensi un doppione, a meno che non siano così superpatiti da non saper rinunciare al possesso di tutte le variazioni sul tema del loro programma preferito. Per tutti gli altri, si presenta un dilemma: meglio *Driller* o *Dark Side*? Dal punto di vista qualitativo chi scrive non può dare un'indicazione netta, ma se è vero che l'originalità è uno degli elementi più importanti di un nuovo programma, dobbiamo preferire senza esitazioni il capostipite del genere: *Driller*.

N.F.R.



BATTLETECH

Il primo "roled war game" elettronico: una grande idea malamente sprecata

Computer: C-64/128
Supporto: Disco
Prezzo: L. 39.000
Produzione: Infocom
Distribuzione: Leader (Via Mazzini 15, 21020 Casciago – 0332/212255)

Nel mondo dei giochi non computerizzati, i prodotti di successo veramente nuovi si possono ormai contare sulla punta delle dita. I tradizionali giochi di società "in scatola" producono solo raramente novità davvero entusiasmanti e sugli scaffali dei grandi magazzini e dei negozi specializzati troviamo soprattutto due categorie di prodotti. Stiamo parlando dei sempre più perfezionati war game giocati su immense mappe esagonate e dei giochi di ruolo che, dopo un iniziale periodo di diffidenza, si stanno ormai diffondendo a macchia d'olio anche fra i più giovani.

Proprio a metà strada fra gli imitatissimi *Third Reich* e *Dungeons & Dragons*, esempi emblematici dei due generi menzionati, esistono alcuni prodotti che riuniscono le caratteristiche di accurata simulazione bellica dei war game e di ampia libertà d'azione dei giochi di ruolo. La novità è stata apprezzata dagli amanti di entrambi i generi, che hanno decretato il successo dei "roled war game".

Battletech, prodotto dall'americana Fasa, è il più significativo rappresentante della categoria, soprattutto per la complessità che ha raggiunto: la serie di prodotti che ne fanno parte è vastissima, e comprende ormai una trentina di moduli separati, che concorrono a formare un universo di gioco dalla coerenza stupefacente. Oltre alla confezione di base per war game (*Battletech*) e al volume di regole per role game (*Mechwarrior*), la Fasa produce espansioni, miniature, sourcebook, manuali tecnici, romanzi e persino una rivista studiata appositamente per coordinare e tenere aggiornate le migliaia di giocatori di *Battletech* sparsi nel mondo.

L'universo così dettagliatamente descritto nelle migliaia di pagine dei manuali che compongono il sistema è situato nel 29° secolo, e si estende tanto da comprendere diverse galassie. Gli innumerevoli sistemi stellari che le compongono sono teatro di sanguinosi e interminabili conflitti armati, che vengono descritti con il nome di Guerre di successione. In altre parole, un'ambientazione che dovrebbe fornire decine e decine di spunti ai programmatori.

Il trono in palio per il vincitore è nientemeno che quello di dominatore dell'universo conosciuto, e i contendenti che lottano per la sua conquista possono essere ridotti a grandi linee a cinque Famiglie, sovente legate da fragilissimi patti di alleanza. Nessun giocatore può interpretare un ruolo di rilievo nella scena politica interplanetaria (tranne che in un'apposita espansione intitolata appunto *Succession wars*, peraltro difficilmente integrabile con gli altri moduli), ma a chiunque è aperta la carriera di combattente in una delle molteplici Armi delle Famiglie.

Ciò che rende *Battletech* più originale sono proprio le inusuali unità che compongono le fila degli eserciti: oltre alle comuni truppe di fanteria, corazzate e aerotrasportate, infatti, i progressi tecnologici hanno permesso la creazione di forze totalmente innovative. A parte il concetto, purtroppo abusato, di battaglie fra astronavi iperspaziali e non, *Battletech* introduce un tipo di veicolo sino a ora trattato esclusivamente dalla fantascienza giapponese dei Manga: il "battlemech". Un battlemech (solitamente abbreviato in 'mech) è un mezzo meccanico di forma umanoide, mosso da un motore a fusione nucleare che permette alla sua massa (variabile fra le 20 e le 200 tonnellate) velocità di crociera paragonabili a quelle delle odierne auto sportive. Questi veloci e corazzatissimi giganti d'acciaio sono inoltre armati con una grande varietà di sistemi d'offesa, che li rendono avversari invincibili per qualsiasi mezzo diverso da un altro 'mech. Il declino della tecnologia e delle risorse seguito ai primi decenni di guerra ha reso queste macchine rarissime, e le poche rimaste vengono affidate soltanto a uomini dalle caratteristiche eccezionali.

Questa lunga introduzione è necessaria per comprendere la logica del programma, primo prodotto elettronico della serie. In *Battletech*, infatti, il giocatore ricopre il ruolo di un apprendista pilota di 'mech, che ha dovuto bruscamente interrompere i suoi studi all'accademia militare per fronteggiare un'incursione nemica sul suo pianeta natale. Lo scopo del gioco, una volta superate le fasi iniziali (nelle quali l'importante è solo sopravvivere), è quello di ritrovare un deposito di 'mech: una missione che aveva impegnato gran parte della vita del personaggio che nella finzione elettronica era il padre del protagonista.

Il programma, che riunisce in sé elementi strategici e di role

game, è controllato interamente tramite joystick e si svolge principalmente su una finestra grafica che occupa due terzi dello schermo, mentre due finestre minori sulla sinistra sono riservate ai messaggi e ai comandi di sistema. I movimenti del joystick controllano un personaggio sullo schermo che, quando ci troviamo a bordo del veicolo, prende l'aspetto di un gigantesco 'mech. Lo scenario, disegnato in un'alta risoluzione cromaticamente infelice, scorre a scatti seguendo la tradizione iniziata anni fa dalla famosa serie *Ultima*.

Dopo pochi minuti di gioco appare ovvio che la maggior parte del divertimento consiste nell'organizzare complesse strategie di

combattimento fra 'mech. Gli elementi di ruolo, purtroppo, vengono pesantemente penalizzati da un limitatissimo sistema di scelte a menu, che per giunta è disponibile solo in locazioni particolari (per esempio all'interno degli edifici). Gli attributi del nostro personaggio sono soltanto tre, e la struttura del programma non permette una vera collaborazione fra i componenti del gruppo di cui facciamo parte nemmeno durante le fasi avanzate del gioco, quando ai nostri solitari vagabondaggi si uniscono gli ex-agenti della squadra speciale di nostro padre.

Il gioco, deludentissimo per la semplicistica realizzazione, giunge alle fasi terminali dopo appena una decina di ore di gioco: un periodo di vita troppo breve per un programma che pretende la qualifica di role game. Queste poche ore bastano comunque per scoprire notevoli incongruenze nella progettazione, che arrivano addirittura a stravolgere lo spirito del gioco originale. La mancanza di comandi flessibili e di un adeguato playtesting hanno portato alla creazione di fasi insostenibili, la cui grossolanità di programmazione può venire sintetizzata in due esempi. 1) I gruppi di giocatori sono graficamente troppo grandi per potersi muovere nelle strade di alcune città. 2) Nella fase finale è necessario dedurre un codice da una serie di circa 50 elementi disposti casualmente, e non viene concesso nessun indizio.

Come nel triste caso di *Ultima V*, la ragione di simili assurdità è da ricercarsi nella pessima scelta da parte della Infocom (famosa per le sue eccezionali avventure testuali) di delegare la programmazione di un prodotto potenzialmente ottimo a un'agenzia esterna, rappresentata in questo caso dalla Westwood Associates, famosa soprattutto per clamorosi fiaschi come *Hillsfar* o *Mars Saga*. I numerosi gadget compresi nella confezione, fra cui spicca il bottone con il simbolo della squadra speciale dei Crescent's Hawks, non bastano per far dimenticare la delusione di un programma che avrebbe potuto essere eccezionale, e che invece si rivela un fallimento sia come role game sia come war game.

Per i numerosi appassionati di *Battletech* non rimane che attendere con pazienza i prossimi prodotti della serie, sperando che vengano curati meglio e consolandosi nel frattempo con gli articoli della rivista *Battle Technology* sulle fantastiche battaglie... che per ora si possono combattere solo sul cartoncino delle mappe esagonate.

F.R.

DRAGON'S LAIR

Una straordinaria animazione che lascia poco spazio al gioco

Computer: Amiga
Supporto: Disco
Prezzo: L. 75.000
Produzione: Visionary Design Tech.
Disponibile presso: Lago (Via Napoleona 16, 22100 Como – 031/300174)

Nel 1983 il mondo dei videogiochi venne sconvolto dalla presentazione di un coin op tanto innovativo da non essere nemmeno paragonabile a ciò che offriva il resto del mercato. Il gioco era dotato di una grafica simile a quella dei migliori cartoni animati e offriva la possibilità di guidare un cavaliere senza macchia ma con parecchia paura nelle numerose stanze di un castello popolato da mostri e trappole di ogni tipo. Non per niente i disegni erano opera di Don Bluth, ex animatore della scuderia Disney e autore "in proprio" del film *Fievel sbarca in America* prodotto da Steven Spielberg. Lo scopo finale era raggiungere una gigantesca camera del tesoro, nella quale il goffo ma potente drago Singe sorvegliava la classica principessa da salvare. Stiamo parlando di *Dragon's Lair*.

Il coin op doveva il suo incredibile aspetto audiovisivo all'impiego di un videodisco laser sul quale erano registrati migliaia di fotogrammi che potevano essere inviati allo schermo in modo non sequenziale. La possibilità di accesso casuale al videodisco permetteva quindi il dipanarsi di una vicenda controllata dal giocatore, che grazie ad appropriate combinazioni di joystick e tasto di fuoco doveva trovare il modo per far procedere senza danni l'eroe, evitando le trappole ed eliminando i vari mostri. Fin dal primo momento, la conversione per home computer di questo gioco è stata

considerata da tutti un'impresa impossibile: la Software Project aveva in effetti pubblicato due giochi ispirati al coin op, ma le limitazioni grafiche, sonore e di memoria delle macchine del tempo (la più potente era il glorioso C-64) avevano costretto a una sintesi tale da rendere i due giochi solo pallidissime imitazioni dell'originale. Poi, arrivò l'Amiga...

Dragon's Lair sul computer a 16 bit di casa Commodore è quantomeno impressionante: alla sola condizione di avere 1 MB di memoria, sul monitor appaiono le sequenze originali del coin op, complete di sonoro digitalizzato. La Visionary Design Technology, realizzatrice dell'incredibile impresa, è riuscita a catturare e a digitalizzare i dettagliatissimi fondali del gioco originale, sui quali sono stati ridisegnati con pazienza certosina (e ottime routine!) gli elementi in movimento, stipando così sui sei dischi di programma informazioni sufficienti a replicare dieci delle 27 stanze dell'originale.

Come nel gioco "al laser", l'azione consiste sostanzialmente in piccoli colpetti dati al joystick nella direzione giusta e al momento giusto, che permettono di far procedere l'animazione sullo schermo nel migliore dei modi. Un movimento sbagliato o fuori tempo provoca il "salto" a una sequenza spesso umoristica nella quale il nostro cavaliere perde una delle sue preziose tre vite. Molte delle dieci stanze (che vengono mostrate sempre nella stessa successione) vengono giocate due volte: per cambiare, il computer le ridisegna invertite specularmente e di conseguenza anche tutte le mosse destra-sinistra risultano invertite.

Quando è arrivato *Dragon's Lair* in redazione, si sono scatenate tali e tante discussioni sul valore di questo prodotto da costringerci per la prima volta a fare una media tra posizioni e giudizi completamente diversi. Il voto sintetico che compare all'inizio rappresenta quindi un compromesso tra ottimo e insufficiente, nel quale hanno il loro peso molte considerazioni che ora cercheremo di esporre con la massima obiettività, lasciandovi l'onore e l'onere di un giudizio definitivo su questo titolo.

Cominciamo con i punti a favore: indubbiamente, *Dragon's Lair* possiede una grafica e un sonoro superiori, assolutamente introvabili in qualsiasi altro gioco per home computer. Lo schema del coin op originale è stato mantenuto con la massima fedeltà, tanto che in alcune stanze gli appassionati della macchina a gettone non avranno problemi a ricordare le complesse sequenze di comandi necessari a superare l'ostacolo.

Dulcis in fundo, esiste la possibilità di ridurre l'area di schermo interessata dalla grafica, ottenendo immagini incredibilmente dettagliate, anche se si deve pagare lo scotto della perdita del sonoro.

Gli aspetti negativi di *Dragon's Lair*, purtroppo, sono altrettanto significativi. Il numero ridotto di stanze, che rende presto monotono il gioco anche a causa della loro successione sempre uguale. In alcune scene, gli elementi grafici mobili più grandi appaiono solo come sagome contornate, declassando quella che sarebbe altrimenti una grafica di altissima qualità. Gli accessi alle sequenze risentono necessariamente della lentezza del disk drive, che viene utilizzato fra una stanza e la successiva e dopo ogni "morte", e persino nel corso della stessa scena, nei casi di animazioni più lunghe e complesse. Il prezzo del prodotto, infine, non può essere certo definito economico.

L'impressione di chi scrive è che *Dragon's Lair*, per quanto rappresenti un'ottima conversione e un prodotto altamente spettacolare, soffra in maniera molto marcata dello stesso difetto che affliggeva il coin op: la limitatezza. Passato l'ammirato stupore iniziale, questo programma si rivela più che altro una grossa demo di grafica e sonoro, dove bisogna muovere il comando in sequenze sempre uguali per poter procedere nella visione. L'interazione è tanto ridotta da mettere in discussione perfino la qualifica di gioco. È un vero gioco quello in cui, come in una catena di montaggio, siamo costretti a seguire le direttive del computer senza che sia possibile mutare la sequenza degli eventi?

Ai lettori l'ardua sentenza.

F.R.

VOYAGER

Bella grafica e molti fronzoli per un poco originale gioco fantascientico

Computer: Amiga
Supporto: Disco
Prezzo: L. 39.000
Produzione: Ocean
Distribuzione: Leader (Via Mazzini 15, 21020 Casciago – 0332/212255)

Il protagonista di questa avventura è Luke Snayles, un criminale di ritorno da un esilio nelle profondità dello spazio durato cinquant'anni. Ma la Terra non è più il verde pianeta che aveva lasciato dieci lustri addietro, bensì un luogo condannato, inerme di fronte alla minaccia di un imminente attacco dallo spazio.

Gli invasori, che stanno costruendo le loro teste di ponte sulle dieci lune di Saturno, sono i temibili Roxiz. Chi conosce gli alieni Vogon, resi celebri dalla famosa *Guida galattica per autostoppisti*, ne ritroverà le caratteristiche nei Roxiz: per gli altri diremo che è sufficiente immaginare un concentrato di crudeltà e malvagità in una forma il più lontana possibile da quella umana per immaginare un Roxiz. Indipendentemente dall'identità degli alieni, il problema dei terrestri sta nel fatto di avere bandito da decenni (siamo nel 2139) le armi più potenti, lasciando così il pianeta indifeso a ogni attacco proveniente dallo spazio.

Fortunatamente la grande astronave nella quale è stato confinato il nostro Luke è dotata di un sistema di armi piuttosto avanzato, che possono essere utilizzate da un modulo esplorativo separabile. Permeato da un senso civico inusuale per uno psicopatico, Snayles decide di utilizzare questa sua relativa capacità di attacco per tentare di porre un fine alla minaccia galattica: il gioco comincia con una spettacolare scena nella quale il modulo armato viene sganciato sulla superficie della prima luna.

Una volta a terra, lo schermo in soggettiva ci mostra uno scenario stupendo, simile a quello di *Starglider 2*, nel quale si aggirano numerosi veicoli Roxiz. Il nostro scopo è di eliminarne una quantità prestabilita, ottenendo così l'accesso a una porta pandimensionale in grado di trasportare il modulo sulla luna successiva, dove le armi dei Roxiz sono più avanzate e pericolose.

Il gioco, in cui l'azione scorre estremamente fluida utilizzando una combinazione di joystick e tastiera, deriva indubbiamente da un vecchio classico del C-64: quell'*Encounter* di Paul Woakes che rimase per un paio d'anni il più veloce gioco in 3D dedicato agli 8 bit di casa Commodore.

Se tralasciamo gli inevitabili miglioramenti grafici resi possibili dall'evoluzione dell'hardware, scopriamo con delusione che nonostante il tempo passato dalla pubblicazione di *Encounter*, i progettisti di *Voyager* non hanno modificato molto la struttura di base. Durante la missione, il nostro veicolo raccoglie alcuni bonus, che hanno effetti diversi sulle sue prestazioni: piramidi che possono essere scagliate come missili dirompenti, scorte energetiche, telecamere, e un modulo "volante" che trasforma il nostro tank in una sorta di hovercraft. L'impiego delle telecamere rappresenta forse l'unica novità del gioco: dovrebbero servire per il lancio di sonde teleguidate dotate di una limitata potenza di fuoco, mentre il modulo principale resta al sicuro in un punto protetto. Quest'opzione, però, si rivela quasi totalmente inutile.

Resta naturalmente una favolosa presentazione con decine di schermate animate riguardanti le lune di Saturno, un sacco di display in sovrimpressione e un aspetto estremamente futuristico. Simpatico ma tanto limitato da avere una longevità prossima a zero, *Voyager* andrebbe consigliato unicamente a coloro che non avendo mai provato l'elegante essenzialità dei suoi progenitori potrebbero lasciarsi ingannare dai fronzoli assolutamente superflui della sua presentazione, non ultima l'audiocassetta inclusa nella confezione. **F.R.**

ULTIMA TRILOGY

Un'antologia storica: i primi tre capitoli della serie Ultima

Computer: C-64/128
Supporto: Disco
Prezzo: L. 59.000
Produzione: Origin
Distribuzione: Leader (Via Mazzini 15, 21020 Casciago – 0332/212255)

Commodore Gazette si è occupata del fenomeno della diffusione dei giochi di ruolo sin dalla loro prima comparsa a livello nazionale, più di un anno fa. Su queste pagine sono stati recensiti decine di role game, appartenenti ai generi più diversi: fantasy, fantascienza, dinamici, grafici, testuali, semplici e complessi, appassionanti e deludenti. Chi ha seguito anche solo parzialmente le nostre recensioni avrà notato i frequenti riferimenti alla saga di *Ultima*, che funge in un certo senso da pietra di paragone per tutti i role game computerizzati. In effetti, la serie di cinque programmi fantasy *Ultima* I, II, III, IV e V, è certamente il ciclo più conosciuto e apprezzato che questo genere abbia mai prodotto.

Alla luce dei numerosi successi ottenuti in termini di critica e di vendite, la Origin di Richard Garriott ha pensato bene di pubblicare in raccolta i primi tre titoli della serie, che nel complesso costituiscono il cosiddetto "ciclo degli stregoni". Sintetizzando il più possibile la vicenda, diremo che i tre dischi contenuti nella confezione danno la possibilità di vivere il primo periodo della storia del mondo di Sosaria, una terra apparentemente medievale governata dal potente Lord British. Le tormentate vicende di Sosaria cominciano con la nascita dello stregone Mondain, cui segue il diffondersi del Male in ogni luogo. Un eroe solitario tenta di sconfiggerlo sgominando le orde animalesche dei suoi servitori e compiendo incredibili imprese persino tra le stelle.

Mondain naturalmente viene sconfitto, ma lascia un erede, Minax, che rappresenta in *Ultima II* una minaccia ancora maggiore per la società sosariana, sconvolta dall'improvvisa comparsa della Magia in ogni luogo. La sconfitta di Mondain si ottiene compiendo lunghe ricerche sotterranee, ma soprattutto grazie a un uso accorto della magia: nel regno di British torna la pace... almeno temporaneamente. Il terzo capitolo di *Ultima* vede la comparsa di un gruppo di eroi, di una stupenda colonna sonora, di una grafica più raffinata e di un avversario misterioso, il tremendo Exodus, che alla fine non si rivelerà nemmeno lontanamente umano.

L'interesse che può suscitare questa antologia di programmi ormai decisamente datati, è di un tipo molto particolare: la grafica estremamente dimessa dei primi due episodi e la limitatissima interfaccia utente impediscono loro di competere, da un punto di vista prettamente estetico, con i prodotti più recenti... ma nonostante siano passati addirittura dieci anni dalla nascita del ciclo, trame e strutture perfettamente bilanciate ne mantengono viva l'attualità in maniera sorprendente. Il terzo programma della serie va considerato poi come un caso a parte, poiché ancor oggi *Exodus: Ultima III* è da considerarsi un modello esemplare nel vasto campo dei giochi di ruolo per quanto riguarda equilibrio, giocabilità, varietà e ampiezza.

La confezione riunisce inoltre la vastissima documentazione dei tre programmi in un unico volume di regole e in una coppia di fogli-mappa... ci sembra che sia un'occasione da non perdere per avvicinarsi al mondo dei giochi di ruolo, anche per l'ottimo rapporto prezzo/longevità offerto dalla raccolta. **F.R.**

WORDPERFECT

Dal mondo dei mainframe e dei PC arriva un elaboratissimo word processor per l'Amiga

Computer: Amiga
Supporto: Disco
Prezzo: L. 773.500
Produzione: WordPerfect Corporation
Distribuzione: Sisoft (Corso Sempione 8, 20145 Milano – 02/33104382)

La prima caratteristica che salta all'occhio di chi si avvicina per la prima volta a *WordPerfect* 4.1 è certamente il suo prezzo vicino alle 800 mila lire. In molti staranno già chiedendosi che cosa può avere di così eccezionale un programma che costa quasi quanto il computer. Il fatto è che questo word processor, a differenza di altri prodotti nati esplicitamente per l'Amiga, è stato "importato" dal mondo MS-DOS, dove costava quasi il doppio. Si sa, nel mondo professionale i prezzi sono diversi, soprattutto perché i prodotti sono destinati a macchine molto più costose e ad aziende per le quali la cosa più importante è l'affidabilità. Con l'Amiga, macchina alla portata di tutti, le cose vanno diversamente e chi produce software deve tenerne conto. Probabilmente la WordPerfect Corporation non poteva scendere più di così.

Abbiamo dunque ottimi motivi per aspettarci che il nostro computer cambi volto in seguito a un investimento tanto consistente. La confezione sembra confermare il nostro ottimismo, in quanto fornisce un contenitore ad anelli con involucro di cartone, uno spesso pacchetto di fogli da inserirvi, un sottomanuale dedicato alla stampa, i quattro dischi del programma e diversi altri gadget come mascherine, prontuari e così via.

All'apertura, il programma ci

propone il consueto schermo del *Workbench* con una finestra in più, quella del word processor. A guardarla bene, questa finestra possiede una caratteristica peculiare: un terzo gadget, in alto a destra, che serve per portarne le dimensioni a tutto schermo in un sol colpo e viceversa (come sul Macintosh). Esiste poi la consueta barra di menu a discesa, molti dei quali si articolano in sotto-menu (tutti dispongono di un'equivalente combinazione di tasti per la loro attivazione). Come vedremo, è possibile assegnare qualsiasi sequenza di comandi a combinazioni di tasti Amiga o Control.

Provando a caricare un file (alla voce Retrieve del primo menu) ci imbattiamo subito in una brutta sorpresa: quando proponiamo come nome del file un asterisco, sperando di ottenere la directory del disco (illusi), l'intero word processor si blocca e dopo un po' anche il resto del sistema. Bene, diciamo, tanti soldi e poi ecco subito un bug. Invece no... o meglio, il bug esiste sul serio, però anche noi abbiamo sbagliato in quanto per leggere la directory si deve usare un'altra opzione, mentre Retrieve serve solo per caricare direttamente un file di cui si conosce il nome.

A ripagarci della disavventura giunge un requester, sottinteso dalla voce List Files, veramente completo: è infatti possibile caricare file, stamparli, eseguire operazioni classiche come la copia o il cambiamento di nome, vedere il contenuto di un file in una finestra ausiliaria senza caricarlo in memoria o anche cercare una parola all'interno di un file sempre senza caricarlo. Ogni volta è possibile utilizzare tanto il mouse quanto la tastiera.

Visto che la complessità del pacchetto sembra notevole, torniamo al manuale e andiamo a vedere come si usa *WordPerfect*. Innanzitutto scopriamo che uno dei quattro dischi contiene una lunga serie di esempi di testi, sui quali si basa la prima parte del manuale per insegnarci a usare un po' tutte le caratteristiche del programma; nella seconda parte, intitolata Reference, vengono invece discusse tutte le opzioni in ordine alfabetico. Alla fine troviamo una sezione dedicata all'installazione, più l'appendice e l'indice analitico con un glossario dei termini più usati nel testo; lo stile tipografico è molto curato, e ogni pagina contiene nel corpo principale la descrizione dei comandi a tastiera, e a fianco una colonna dedicata alle indicazioni sul modo d'impartire gli stessi comandi via mouse.

Riusciamo quindi a caricare in memoria uno dei file del disco Learn, e sullo schermo compare il testo redatto con diversi stili e formati. *WordPerfect* è fatto in modo da mostrare all'utente il testo così come sarà stampato, compresi i cambiamenti di stile e di allineamento e le eventuali annotazioni a piè di pagina. In realtà alcune cose, come la modalità Superscript e le frasi riportate in tutte le pagine in alto o in basso (l'*header* e il *footer*), non vengono visualizzate. A questo si può però ovviare passando al modo Reveal Codes (Alt-F3), che visualizza il testo in una finestra a parte, evidenziando tutti i caratteri di controllo (oppure si può ridirigere la stampa su un file o su una finestra CLI). La stampa può, a dire il vero, essere dirottata su qualsiasi dispositivo, compresi il PIPE: o lo SPEAK:, e anche questa è una caratteristica insolita e notevole per un word processor.

Lo spostamento del cursore avviene nel solito modo: attraverso il mouse o mediante la barra di scroll presente sulla destra. Ci si può muovere di parola in parola o a inizio e fine riga mediante le consuete combinazioni di tasti CTRL-qualcosa, e si possono effettuare "salti" più consistenti ricorrendo alla tastierina numerica che *WordPerfect*, a differenza di molti altri programmi, rende attiva anche per le funzioni non numeriche; i blocchi si possono selezionare mediante il mouse (con lo scroll automatico del te-

sto), oppure mediante comandi abbreviati che sveltiscono la selezione di righe o paragrafi. A volte le sottovoci dei menu a tendina permettono di svolgere una determinata operazione direttamente sulla frase o sulla riga indicata dal cursore, scavalcando la necessità di definirla. È anche possibile definire blocchi rettangolari (a colonna).

Le funzioni di ricerca, sostituzione e cut & paste operano nel solito modo: diciamo soltanto che non è previsto l'uso di caratteri jolly nella ricerca delle parole, ma solo la corrispondenza o meno di maiuscole e minuscole. Invece, sempre dal secondo menu, scopriamo che esistono ben tre livelli di Undo (validi per ciascuno dei metodi di cancellazione utilizzati: tasto backspace, delete o le funzioni di "cancella fino a..." dei menu). Per ogni livello di Undo viene generato un file temporaneo; a tale proposito osserviamo che possono essere generati file temporanei anche per il periodico backup dei file o per la manipolazione di testi che superano la memoria disponibile.

A un certo punto scopriamo anche che esiste un help in linea, che può essere attivato premendo il tasto Help e poi la lettera iniziale del comando, oppure il tasto funzione, di cui vogliamo conoscere il significato.

Dal menu Format si possono impostare tutti i parametri del foglio di stampa. Abbiamo a disposizione due header e due footer (A e B), le note a piè di pagina, le note a fine testo, la numerazione con eventuali rientranze successive dei paragrafi, la numerazione delle pagine, nonché la possibilità di spezzare il testo in più colonne con diverse modalità. Si possono poi variare i parametri di stampa, anche se per la stampa esiste un sofisticato menu che si potrebbe quasi considerare un task a sé.

Quando decidiamo di stampare qualcosa, infatti, (blocco, pagina o l'intero file), ci occorre anzitutto il disco Print, poi dobbiamo dialogare con un vero e proprio "pannello di controllo" tramite il quale si può scegliere il driver per la stampante, il dispositivo di output, il tipo di trascinamento, i nomi dei file da accodare in stampa e così via; dopodiché viene attivato un task secondario (uno per ogni operazione di stampa richiesta), il cui stato e la cui priorità sono aggiornati sullo schermo del pannello di controllo. Possiamo interrompere o alterare l'operazione di stampa quando vogliamo, ed eventuali problemi ci verranno sempre segnalati in questo pannello.

Il menu successivo, Style, è quello delegato alla scelta degli stili dei caratteri: le varie opzioni agiscono sul testo dal punto in cui si trova il cursore, oppure possono essere usate per conoscere gli stili di un testo già scritto o per alterarli. Anche qui hanno valore le consuete operazioni per la definizione dei blocchi, anche se stranamente non è possibile utilizzare la selezione rapida, per esempio delle frasi o dei paragrafi. Inoltre non si possono memorizzare separatamente, per utilizzarli in occasioni successive, i soli stili e formati.

Nell'ultimo menu, Special, troviamo poi una sequenza di vere e proprie leccornie, una più piacevole dell'altra. Si comincia dallo spell-checker e dal thesaurus, rispettivamente controllore dell'ortografia e dizionario dei sinonimi. Tramite queste due opzioni è possibile controllare e sottoporre a verifica ogni parola mentre viene scritta oppure l'intero testo già, scritto, o anche cercare sinonimi (sia sostantivi sia verbi) per le parole indicate. Si possono insegnare al vocabolario nuove parole, che verranno riconosciute al pari delle altre.

Oltre a questo, *WordPerfect* ci permette anche d'indicare il modo di spezzare le parole, proponendoci una scelta di default e consentendoci di cambiarla; si può anche indicargli che una parola composta non va spezzata o che un paragrafo deve restare tutto nella stessa pagina. L'unico problema con queste opzioni è che sono state create per la lingua inglese, e quindi per noi sono di ben poca utilità; solo per la tastiera possiamo arrangiarci in qualche modo ridefinendo alcune combinazioni di tasti Amiga e/o CTRL per ottenere i caratteri tipici della nostra lingua. In ogni caso possiamo sempre utilizzare il comando SETMAP dell'AmigaDOS.

Di meraviglia in meraviglia, troviamo poi il mail merge per la stampa delle circolari o delle etichette, che ormai è diventato obbligatorio per tutti i word processor di un certo livello; da parte sua *WordPerfect* aggiunge alcune metaistruzioni per bloccare la sostituzione automatica dei campi e richiedere input da tastiera, per invocare macro (con le quali la sostituzione può essere resa condizionale) e altro ancora. E vediamo finalmente queste macro: si tratta di sequenze di scelte dai menu a tendina o di comandi comunque impostati, che vengono memorizzati sotto un nome specifico o in una combinazione Amiga-qualcosa. A questo scopo basta attivare il modo Def affinché il computer inizi a memorizzare i comandi a mano a mano che vengono impartiti; alla fine si disattiva il modo di registrazione e la serie di comandi può essere ripetuta con una sola operazione.

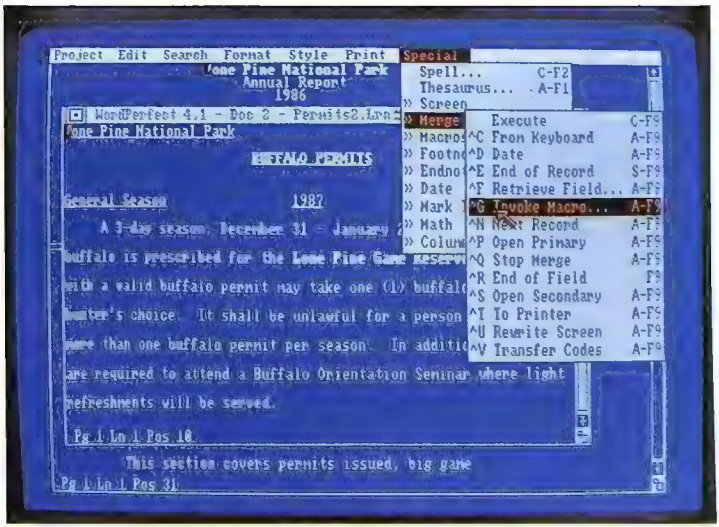
Per mezzo delle macro (visto che si può anche creare una catena, condizionale o meno, di macroistruzioni) è in pratica possibile creare semplici programmi per il trattamento dei testi. Bisogna solo fare attenzione a non farle passare attraverso il CLI: come c'era infatti da aspettarsi, il word processor non può mante-

nere il controllo del sistema dall'esterno (ossia dal livello CLI) e provando a farlo a noi sono capitate cose strane... selezione indesiderata di opzioni che non erano state richieste, anomalie nella gestione dei file e infine Guru Meditation.

Si può poi definire in dettaglio una struttura di testo a più colonne, e infine si può passare al modo Math, per l'esecuzione di calcoli. In questo ambito, la parte del foglio delimitata dagli identificatori [Math] (codici analoghi a quelli di formattazione) si trasforma quasi in un tabellone elettronico, permettendo di definire alcuni campi su cui vanno effettuate le operazioni. Si possono avere infatti campi di testo, campi numerici e campi contenenti formule: i campi numerici possono essere usati come operandi per somme, sottrazioni, moltiplicazioni e divisioni, nonché medie aritmetiche o geometriche. Esiste anche la possibilità di ottenere totali parziali e generali.

Riassumendo, diciamo anzitutto che va fatto un grande elogio a *WordPerfect* per essere il primo word processor che fa un uso esteso e costante delle potenzialità dell'Amiga (multitasking, finestre...). Finalmente si possono tenere in memoria più file e tuttavia continuare a lavorare come se ce ne fosse uno solo, e si può attivare la stampa di più file senza impegnare il word processor. Tutto questo, ovviamente, entro i limiti hardware dell'Amiga.

Il programma possiede inoltre innumerevoli opzioni che coprono ogni esigenza, opzioni che come abbiamo visto spaziano dalla definizione del testo alla definizione del formato di pagina, a elementi collaterali che però in certi casi possono essere utili. Sicuramente non abbiamo parlato di tutto ciò che offre il programma, ma aggiungiamo ora, solo per fare un esempio, che è possibile rendere illeggibile un file agli estranei inserendo una password (cosa che ne provoca automaticamente la crittografazione). Alle poche mancanze, come per esempio la ridefinizione della tastiera o la fusione tra testo e grafica si sopperirà probabilmente con la versione 5.0, al momento già disponibile per altri sistemi operativi.



Piuttosto, sono da notare alcune incongruenze, a nostro parere dovute a sviste nella programmazione durante il passaggio al mondo Amiga. Ci riferiamo a certe opzioni dei menu a tendina, che spesso sembrano incoerenti o incomplete: per esempio, è esplicitamente possibile forzare l'allineamento a destra mentre quello a sinistra esiste solo come default; in alcuni casi non si riescono a riportare tutti i caratteri di un blocco allo stile di default, anche se il menu afferma che l'operazione è stata eseguita. Probabilmente la presenza dell'interfaccia user-friendly ha confuso un po' le idee ai programmatori di *WordPerfect*, che alla fine hanno tralasciato qualcosa.

Vista la quantità di opzioni e la complessità del programma, non c'è da stupirsi che qualche problema non sia stato risolto o sia magari scaturito involontariamente nella conversione da un mondo all'altro... ricordiamo infatti che pur trattandosi della versione 4.1, questa è in realtà la prima versione disponibile per l'Amiga di un prodotto che ha alle spalle quasi un decennio di esperienza nei mondi IBM e Digital. Se il prezzo sembra eccessivo, oppure se si ritiene che per una simile cifra si deve avere il diritto di acquistare un prodotto del tutto privo di bug e di mancanze, ci si può rivolgere ad altri buoni word processor dal costo cinquedieci volte inferiore.

Però è facile constatare che *WordPerfect* è un prodotto praticamente unico per il numero di possibilità che offre, e vi sono alcuni utenti, soprattutto professionali, che hanno un assoluto bisogno di uno spettro di opzioni "a tutto campo". Se acquistano *WordPerfect* difficilmente dovranno pentirsene. **D.G.**

COSMIC PIRATE

All'arrembaggio! Arriva uno dei più movimentati e fantasiosi giochi spaziali degli ultimi tempi

Computer: Amiga
Supporto: Disco
Prezzo: L. 39.000
Produzione: Outlaw
Distribuzione: Lago (Via Napoleona 16, 22100 Como – 031/300174)

Che cosa può spingere un uomo a diventare un pirata dello spazio?

Troppe repliche di *Capitan Harlock* in TV, per esempio, ma nel caso del protagonista di questo gioco la ragione è molto più grave: tanto grave che nemmeno il manuale delle istruzioni osa parlarne. Al contrario di altri giochi in cui i programmatori si divertono a inventare improbabili vicende fantascientifiche di cui poi non si trova traccia nella vicenda elettronica, in *Cosmic Pirate* non veniamo a conoscenza di nessun complesso retroscena.

In effetti, il background di *Cosmic Pirate* non è particolarmente importante: quel che conta è sapere che una volta caricato il programma ci troveremo a far parte del NEST 51, alle dipendenze della più grande organizzazione criminale dell'universo conosciuto. Il NEST è una sorta di sindacato che cura gli interessi dei fuorilegge intergalattici, fornendo loro vitto, alloggio, armi, mezzi e preziosissime informazioni riguardanti le posizioni dei cargo spaziali da abbordare.

Essendo l'organizzazione formata esclusivamente da fuorilegge, tuttavia, non c'è da aspettarsi una serena atmosfera di cameratismo. I servizi forniti dalle grandi basi del NEST 51 vengono pagati a peso di platino, e i pirati alle prime armi possono facilmente vedersi recapitare fatture spaventose che, trattenendo sino al 99 per cento del bottino, hanno spesso l'effetto di far andare in rosso il loro bilancio. Se i poveri pirati avevano scelto la loro nuova carriera per sfuggire ai taglieggiamenti legali delle imposte dirette, a questo punto devono constatare che la musica è sempre la stessa. Nemmeno tra i fuorilegge possono sperare in un po' di umanità! La chiave per sopravvivere riuscendo a trattenere per sé grosse fette del maltolto sta tutta nel numero di missioni completate con successo: migliorando l'opinione che il NEST ha di noi abbiamo la possibilità di ottenere grossi sconti rispetto alle tariffe di base.

I guadagni, che aumentano a seconda della difficoltà della missione e della nostra anzianità, possono essere utilizzati in vari modi. Il modo migliore di spendere i crediti accumulati è senz'altro rappresentato da investimenti per aumentare le prestazioni della nostra nave: l'interno dell'organizzatissimo pianeta-fortezza del NEST 51 a cui facciamo capo offre ottime (e costosissime) opzioni per migliorare le prestazioni del nostro mezzo.

Motori, laser e scudi di varia efficacia possono essere acquistati direttamente sulla rampa di lancio per andare a migliorare in maniera permanente le caratteristiche di manovrabilità, attacco e difesa dello sguarnito caccia di forma sferoidale che ci viene offerto all'inizio del gioco. Il potenziamento dell'astronave è fondamentale per sopravvivere nelle missioni di livello avanzato, che siamo obbligati ad affrontare col procedere della nostra carriera.

Oppure i crediti si possono spendere nei "simulatori": fra una missione e l'altra abbiamo infatti libero accesso all'equivalente spaziale di una sala giochi, dove troviamo numerosi programmi simili a videogame (i simulatori, appunto) che permettono di esaminare in forma schematica, e soprattutto innocua, i pericoli che dovremo prima o poi affrontare nel corso del gioco. Stranamente alcuni sono gratuiti, e servirsene tempestivamente è particolarmente consigliabile a chi abbia trovato difficoltà nello svolgimento di una missione "reale".

Ma come si svolgono le missioni di cui continuiamo a parlare? Qual è in realtà il lavoro quotidiano di un pirata cosmico?

Una tipica missione offerta dal NEST 51 consiste nell'arrembaggio di un grosso cargo spaziale, che dev'essere reso inoffensivo utilizzando le armi di bordo e quindi agganciato e trainato sino al NEST. Il carico viene subito confiscato dall'organizzazione e una piccola percentuale del suo valore finisce sul conto di colui che ha portato a temine con successo l'impresa. Generalmente, prima di poter raggiungere le navi mercantili è necessario attraversare diversi settori galattici, affollati di pericolosi avversari.

Gli ostacoli che ogni pirata spaziale deve affrontare e superare sono costituiti soprattutto da miriadi di formazioni aliene fermamente intenzionate a eliminare la pirateria spaziale. Questo sciagurato proposito si traduce in furiosi assalti a colpi di laser o addirittura in attacchi suicidi delle astronavi nemiche, che hanno l'effetto di diminuire la potenza degli scudi di bordo del povero pirata cosmico. Una volta che gli scudi (capaci comunque di ricaricarsi automaticamente in assenza di attacchi nemici) sono stati completamente abbattuti, comincia a crescere sullo schermo la barra che segnala i danni subiti: quando la barra raggiunge la lunghezza massima il caccia pirata dev'essere considerato distrutto e il pilota viene riportato al NEST, e si ritrova addebitato sul suo conto il costo di una nuova navicella.

Oltre agli ostacoli attivi, che possono essere eliminati con una

buona scarica di laser, c'è il problema delle dogane spazio-temporali. Queste strutture, costituite da una sorta di barriera circolare che circonda un buco nero, vengono utilizzate per passare da un settore all'altro, ma possono essere attraversate solo pagando un pedaggio variabile da 0 a 7 mila crediti. Inutile a dirsi, i crediti in questione non possono essere prelevati dal proprio conto al NEST 51, e vanno perciò accumulati distruggendo le già citate formazioni di astronavi. Fra i relitti dei mezzi nemici si possono a volte trovare utili bonus che aumentano i crediti disponibili o hanno altri piacevoli effetti, fra i quali la riparazione di parte dei nostri danni, la distruzione di tutti i nemici presenti, l'incameramento di una bomba intelligente o altro.

Settore dopo settore, raggiungiamo finalmente il cargo. Ora tutto quello che dobbiamo fare è inseguirlo e neutralizzarne le parti in movimento (utilissime in questo caso le bombe intelligenti) per poterlo finalmente agganciare e trasportare al NEST. A seconda della difficoltà della missione cambiano le dimensioni del cargo, che possono addirittura estendersi per diverse schermate, pullulanti di motori, cannoni e diavolerie assortite.

Una volta che il cargo è stato agganciato, il viaggio di ritorno consiste in un velocissimo quadro bonus nel quale numerose e microscopiche astronavine compiono spettacolari acrobazie attorno al nostro mezzo: abbattendole incameriamo crediti addizionali che, una volta detratte le immancabili tasse, vanno ad aggiungersi al nostro patrimonio faticosamente accumulato.

L'ultima possibilità che abbiamo per mettere insieme un tesoro che ci renda degni del nome di pirati, ci viene offerta dalle visite ai pianeti, da cui possiamo ottenere lauti guadagni. In queste fasi il giocatore affronta forze aeree dotate di bombe a scoppio ritardato, insidiosissime ma adeguate alla ricompensa che otteniamo per la loro eliminazione.

Cosmic Pirate, realizzato per la Outlaw (quella del *S.E.U.C.K.*) dalla Zippo Games, nota software house nipponica dedita principalmente alla realizzazione di cartucce per console, rappresenta una vera eccezione nel vasto panorama di videogiochi per l'Amiga. In questo gioco, infatti, gli aspetti grafici e sonori vengono lasciati in secondo piano rispetto alla velocità dell'azione e alla giocabilità, che risultano elevatissime. Video e audio, sia chiaro, sono accettabilissimi, e risultano anzi superiori a quelli offerti da molti altri prodotti, ma ciò che cattura l'interesse in *Cosmic Pirate* è la facilità con la quale si entra nello spirito del gioco, aspetto troppo spesso sottovalutato nel software a 16 bit.

La grande giocabilità è favorita dalla amichevolissima interfaccia utente, che riesce a rendere immediatamente accessibili e comprensibili le numerose opzioni offerte dal NEST 51, in una forma colorata e affascinante.

Durante tutto lo svolgersi del gioco, appaiono sul nostro schermo vere e proprie gag: astronavi a forma di pesce che diventano sempre più grandi nelle fasi di combattimento, scorrerie che ci fruttano un carico di 800 mila tonnellate di calzini (marroni) diretti ai magazzini di W. H. Smith...

Non manca fra i serissimi simulatori in grafica vettoriale una versione futuristica del preistorico *Pong*: nonostante il tremendo "costo" nessun appassionato riuscirà a resistere alla tentazione di farsi una partitina o due, intaccando magari il proprio scarso capitale in maniera irreparabile.

Cosmic Pirate, reso ancora più godibile dalla caratteristica più unica che rara di salvataggio automatico della posizione raggiunta e dalle simpatiche istruzioni tradotte in italiano, è un prodotto interessantissimo che consigliamo a tutti gli appassionati dei giochi d'azione, e che non ci stupiremmo di trovare in futuro fra i classici.

F.R.

MANHUNTER: NEW YORK

Un gioco a due facce: grande fantasia e varietà d'azione, e grafica/sonoro/animazione quasi inesistenti

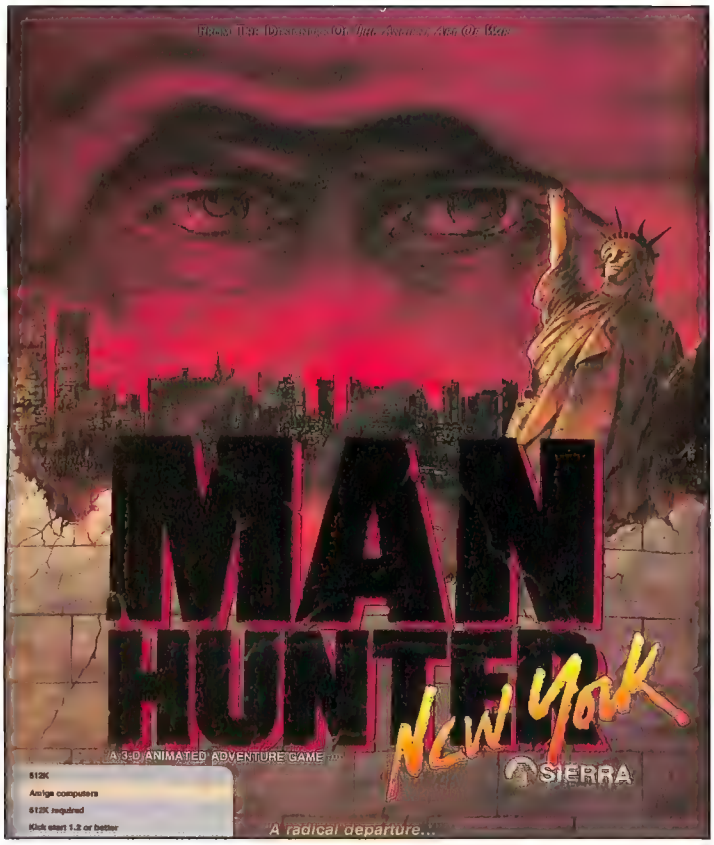
Computer: Amiga
Supporto: Disco
Prezzo: L. 59.000
Produzione: Sierra
Distribuzione: Leader (Via Mazzini 15, 21020 Casciago – 0332/212255)

L'anno è il 2004: la Terra viene invasa da una strana razza aliena che in brevissimo tempo riesce a conquistare il potere instaurando una spietata dittatura. Gli esseri umani vengono relegati in "città di concentramento", zone dalle quali non è possibile uscire e dove la posizione di ciascun individuo è costantemente controllata da un apparecchio installato bionicamente del quale non ci si può liberare. Gli apparecchi di controllo, tuttavia, non sono perfetti: se il soggetto che li indossa si sposta al di sotto del livello del mare, diventa virtualmente invisibile agli strumenti di controllo alieni, e in ogni caso i segnalatori non sono in grado di comunicare l'identità dell'organismo su cui sono stati installati.

Come in ogni regime imposto contro la volontà della popolazione, presto nelle città si sviluppano moti rivoluzionari che ottengono subito una larga adesione, preoccupando non poco i leader alieni. Questa situazione porta alla nascita di corpi speciali di collaborazionisti battezzati "manhunter", il cui compito è investigare sulle attività di persone sospette infiltrandosi fra loro. Agendo nella massima segretezza, i manhunter hanno a loro disposizione un apparecchio computerizzato chiamato MAD, in grado di collegarsi alla rete di computer aliena e che quindi permette loro d'individuare i segnali inviati dai congegni identificatori individuali.

L'azione di *Manhunter: New York* si svolge nella famosissima isola di Manhattan, l'unica zona della città nella quale gli uomini hanno il permesso di vivere. Nei panni di un manhunter veniamo svegliati nel cuore della notte per investigare su un attentato dinamitardo avvenuto all'ospedale cittadino: nottetempo qualcuno ha demolito una parete dell'edificio ed è penetrato nell'obitorio dell'ospedale.

Una rapida occhiata al nostro MAD ci permette di scoprire quali sono stati gli spostamenti del bombarolo nelle ultime ore: tutto quello che dobbiamo fare è seguirne le tracce per poterlo identificare e catturare. Partendo dall'ospedale, scopriamo che l'attentatore ha frugato la salma di un manhunter deceduto in circostanze misteriose: il divieto assoluto di comunicare con gli altri esseri umani imposto dalla dittatura aliena ci impedisce di ottenere altre informazioni, così non ci resta altro che passare alla tappa successiva.



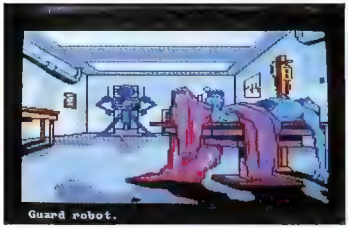
Seguendo i segnali registrati dal nostro MAD ci avventuriamo in un fetido bar nel quale dobbia-

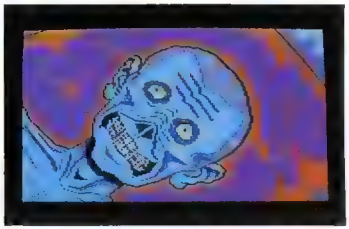
mo affrontare la follia omicida di un gruppo di colossi armati di coltelli, poi passiamo in una chiesa abbandonata deserta e in altri luoghi, finché l'inseguimento non si conclude nelle sporchissime latrine di un parco pubblico, dove il MAD aveva registrato l'ultimo segnale. Possibile che la nostra preda sia scesa nel sottosuolo proprio da lì? E come?

Altri indizi ci portano nello squallido luna park di Coney Island, dove scopriamo facilmente un sistema di segnali legato a particolari giochi d'abilità... e dove inizia la seconda parte dell'inseguimento, che si concluderà in maniera inaspettata e decisamente spettacolare. Per i meno esperti è previsto un piccolo aiuto: una sezione del manuale, intitolata "Manhunter Walk-Thru", fornisce indicazioni a volte preziose su come destreggiarsi nelle prime fasi del gioco. Gli avventurieri di lunga data non dovrebbero averne bisogno... ma non siatene troppo sicuri.

Manhunter: New York è stato programmato dalla Sierra (ex Sierra-On-Line), la stessa software house responsabile del contraddittorio *Police Quest* recensito nello scorso numero di *Commodore Gazette*. Com'è abitudine per i programmatori della Sierra, anche questo gioco è stato originariamente sviluppato per i PC IBM, e ancora una volta questa genealogia si rivela deleteria. Pur non arrivato alla grafica abissalmente primitiva di *Police Quest*, anche l'aspetto di *Manhunter* risente pesantemente delle limitazioni dell'ambiente in cui è stato sviluppato. In pratica, seguendo una triste politica di conversioni "pixel per pixel", ancora una volta la Sierra presenta sull'Amiga un bellissimo programma... per IBM! Dopo le spettacolari visioni dei vari *Dragon's Lair* e dei titoli *Cinemaware*, la grafica a blocchi di *Manhunter* si rivela addirittura scioccante nella semplicità delle sue animazioni costituite in media da tre quadri grafici.

Sotto l'aspetto del sonoro, i quattro canali audio del computer sono sfruttati unicamente per emulare il cicalino piezoelettrico delle macchine IBM, con sgradevoli effetti "pernacchianti", e lo stesso discorso va fatto per il sistema di comando. Nonostante i programmatori si siano prodigati in titanici sforzi per rendere possibile l'uso del mouse, il miglior modo per giocare a *Manhunter* rimane l'uso costante ed esclusivo della tastiera, che assicura risposte più rapide e affidabili di quelle offerte dall'ormai diffusissima "scatoletta magica".


Guard robot.



Dopo tutte queste osservazioni indubbiamente negative, dovremmo proprio bollare quest'ennesima avventura dinamica come un fallimento... e invece la questione è tutta da discutere.

Come abbiamo già osservato nella recensione a *Dragon's Lair*, l'esistenza di due categorie ben distinte di giocatori impedisce di dare un giudizio univoco. Chi ha comprato l'Amiga solo ed esclusivamente per bearsi della sua grafica e del suo sonoro superiore dovrebbe senz'altro evitare prodotti come *Manhunter* o altre avventure marchiate Sierra, dal momento che in molti casi l'aspetto di questi programmi potrebbe essere considerato negativo anche in un C-64 (che, detto fra noi, continua a essere un ottimo computer!). Tutti gli altri, cioè coloro che in un gioco cercano soprattutto il divertimento, e che identificano il divertimento con la varietà delle situazioni possibili e la complessità del gioco, farebbero invece molto bene a dare almeno una rapida occhiata a questo titolo.

Caricando *Manhunter* tornano subito in mente i primi programmi per l'Apple: titoli come *The Prisoner*, che oggi sono quasi dimenticati ma che hanno dato origine a un'intera categoria di programmi di cui oggi non si contano più gli imitatori. In quei giochi, proprio come in questo *Manhunter*, ci si trovava di fronte a innumerevoli situazioni diverse, tanto che era difficile trovare una definizione schematica per inquadrare il programma in un genere preciso.

Esattamente la stessa osservazione si può riferire a *Manhunter*, definito "adventure game" nella pubblicità della Sierra, ma molto più vario e ricco di quello che ci si aspetterebbe da un'adventure qualunque. Nel corso della missione s'incontrano enigmi polizieschi, labirinti, tiri al bersaglio, opere di ricerca sulla documentazione fornita e molte altre cose ancora, in uno schema di gioco in continua evoluzione.

Il nostro consiglio è di provare anche prodotti come questo: chissà che non scopriate un nuovo tipo di divertimento, che si può apprezzare anche senza un monitor ad altissima definizione...

F.R.

TIME SCANNER

Un complicato flipper a 16 bit... che non corre il rischio di andare in "tilt"!

Computer: Amiga
Supporto: Disco
Prezzo: L. 49.000
Produzione: Activision
Distribuzione: Leader (Via Mazzini 15, 21020 Casciago – 0332/212255)

Forse per la sua convivenza forzata accanto ai coin op, il flipper è stato sin dai tempi dei primi videogame l'ispiratore di parecchi giochi elettronici. Molti utenti del Commodore 64 ricorderanno certo ancor oggi il primitivo ma giocabilissimo *David's Midnight Magic*, e non per caso due software house oggi famosissime, la SubLogic e la Electronic Arts, hanno ottenuto i loro primi successi commerciali proprio con giochi ispirati ai flipper, rispettivamente *Night Mission* e *Pinball Construction Set*.

Dopo alcuni anni di relativa stasi nel campo delle riproduzioni digitali del mitico passatempo del "pinball", in Italia chiamato erroneamente "flipper" a causa delle scritte che sovrastavano le palette mobili dei primi modelli, la Activision lancia ora un nuovo biliardino elettronico, con questa sua conversione di un coin op pressoché sconosciuto in Italia prodotto dalla software house giapponese Sega.

Prima di cominciare la recensione del programma, vale la pena di ricordare un divertente aneddoto legato a questo gioco: la pubblicazione del programma è stata rimandata diverse volte, per le ragioni più strane. Se inizialmente i motivi di questo ritardo sono stati la priorità da accordare ad altre conversioni o la palese incompetenza dei programmatori a cui era stato commissionato il lavoro, l'ultimo evento che ha ritardato l'uscita della versione per home computer sembra addirittura farsesco.

Il giorno in cui la presunta versione definitiva di questo programma venne presentata al general manager della Activision, raccontano le voci di corridoio, i responsabili della conversione furono licenziati in tronco, nonostante la strabiliante e perfetta trasposizione della grafica e del sonoro originali... perché non avevano nemmeno provato a programmare realistiche routine di rimbalzo della pallina. La biglia d'acciaio levitava eterea sul piano di gioco, compiendo rapidi scatti casuali da e verso i bumper, gestita da routine vincolate così strettamente al resto del programma da rendere impossibile qualunque modifica. La Activision fu costretta a ordinarne un'ennesima versione.

Quanto ci sia di vero in questa storiella non possiamo indovinarlo, ma bisogna riconoscere che fatti come questo sono quasi all'ordine del giorno nel mondo tutto particolare dei programmatori di giochi, e ci è capitato di sentire anche di peggio. Ricordate la complicata storia di *R-Type* e *Denaris*?

Lasciando i pettegolezzi da rotocalco, dobbiamo comunque ammettere che la lunga gestazione di *Time Scanner* ha partorito alla fine una conversione di ottima qualità, praticamente indistinguibile dal coin op della Sega persino sotto l'aspetto del sonoro e della presentazione. L'unica cosa di cui si sente la mancanza è quel particolare "modulo di comando" che nell'originale era montato sul cassone del gioco, a immagine e somiglianza del pianale di un flipper (mi si perdoni l'espressione scorretta), che era possibile scuotere e malmenare proprio come un tavolo completo.

Come ben sanno gli esperti del nobile gioco, dietro alla meccanica piuttosto semplice di queste macchine si nascondono spesso complicati schemi di gioco, e anche nella versione elettronica esistono complesse strategie, anche se viene rispettata la filosofia di base del flipper: buche, rimbalzi, corridoi e bersagli da abbattere.

Subito dopo il caricamento, ci troviamo su di un ripiano chiamato "Volcano", di cui vediamo soltanto una metà per volta; l'obiettivo è abbattere una serie di

tre bersagli per liberare l'accesso a una rampa sopraelevata che, se viene imboccata a un'adeguata velocità, provoca "l'eruzione" di una delle lettere che compongono il nome del gioco. Una volta completata la parola "Volcano", sul piano di gioco vengono liberate due palline addizionali che ci permettono d'incrementare ulteriormente il punteggio raggiunto.

Ma *Time Scanner* non si limita a proporre un gioco che fino a questo momento sarebbe realizzabile anche meccanicamente. Per ogni buon programma ludico che "imita" qualcosa di già esistente è quasi obbligatorio fornire qualche elemento aggiuntivo spettacolare e caratteristico del mezzo elettronico. Diligentemente, *Time Scanner* rispetta quest'obbligo morale mettendo a disposizione del giocatore una buca chiamata "Time Travel", che provoca il caricamento di un nuovo pianale.

Il nuovo gioco, intitolato "Ruins", ha la caratteristica di possedere nella parte alta una buca che conduce le palline a un contenitore trasparente situato sotto il piano di gioco. Se si riesce a riempire il contenitore con tre palline, in cambio si ottiene il multi-ball e un grande bonus. L'ultima delle tre palline però è una speciale "Fire Ball", che si può ottenere soltanto abbattendo una serie di bersagli.

Anche qui esiste la possibilità d'imboccare un passaggio Time Travel, che conduce al terzo pianale, ambientato fra le piramidi egiziane e chiamato "Saqqarah". Si tratta dello schema di gioco più complesso, che richiede l'abbattimento di una serie di bersagli indicati dalle lettere "P,y,r,a,m,i,d" e l'introduzione di tre palline in un tunnel contorto e di difficile accesso.

Ma non è finita: il Time Travel esiste anche nel Saqqarah, e conduce o al "Volcano" o a un quarto pianale, sul quale si può arrivare solo a condizione di aver soddisfatto gli scopi dei tre pianali precedenti. L'ultimo schema, chiamato genericamente "Special", non ha uscite Time Travel, ma possiede nella parte alta una zona rettangolare il cui pavimento è composto da un gran numero di bersagli a caduta. All'inizio i bersagli formano la lettera S in rilievo, e noi dobbiamo abbatterli cancellando la lettera. Una volta soppressa la S, i bersagli si rialzano prendendo la forma di una P, e così via sino a scrivere la parola SPECIAL, indicata anche da una serie di lampade luminose sul pianale. Quel che accade una volta completata la strage di bersagli è una sorpresa che non riveliamo... possiamo solo dire che è molto redditizia in termini di punteggio.

Oltre alle caratteristiche descritte, *Time Scanner* possiede naturalmente tutti i gadget di un comune flipper, e comprende bumper, ponti, flipper supplementari, buche, molle, magneti e chi più ne ha più ne metta: le partite procedono in una cacofonia di suoni, lampi e luci colorate che restituiscono piuttosto bene l'atmosfera di una sala giochi. L'accompagnamento è completato da quattro diversi motivi musicali che corrispondono ai vari piani di gioco.

Prima di concludere vanno rilevate ancora due caratteristiche di *Time Scanner*: la possibilità di continuare a giocare dalla situazione raggiunta, mantenendo lo stato dei vari bonus, e l'impossibilità di mandare il gioco in "tilt", favorendo così partite tutte colpi e spintoni, gestiti ottimamente sia dal mouse che dalla tastiera.

Un acquisto indispensabile per gli amanti della biglia argentata (che però qui è verde), che potrebbe costituire un'ottima fonte di divertimento anche per i fanatici degli shoot'em up.

F.R.

SILKWORM

Un frenetico shoot'em up che non ha nulla da invidiare al suo gemello delle sale giochi

Computer: C-64/128
Supporto: Cassetta/Disco
Prezzo: L. 18.000/21.000
Produzione: Virgin Games
Distribuzione: Leader (Via Mazzini 15, 21020 Casciago – 0332/212255)

Sintetizzando la lunga e leggermente confusa storia raccontata dal foglio d'istruzioni di *Silkworm*, possiamo dire che la missione "Baco da seta" ("silkworm", appunto) consiste in un raid suicida contro la più vasta armata mai concepita da mente umana. L'obiettivo è la distruzione completa delle forze corazzate che si muovono verso di noi, nel tentativo di scongiurare un violentissimo conflitto che porterebbe alla distruzione completa del nostro pianeta.

Seguendo una piacevole tendenza dei più recenti coin op, la conversione per home computer di *Silkworm* dà a due giocatori la possibilità di affrontare contemporaneamente la sfida offerta dalla missione, utilizzando veicoli differenti. Come già avevamo visto in *Last Duel* della Capcom, recensito nel numero scorso, i mezzi a disposizione dei giocatori sono complementari: uno agisce nell'aria e l'altro sulla terra. Prima dell'inizio della partita è possibile scegliere se il proprio veicolo sarà una jeep oppure un elicottero.

Il mezzo terrestre, dotato di un lanciamissili che può essere inclinato entro un arco di 180 gradi, fa tornare alla mente reminiscenze di quel coin op del 1983 intitolato *Moon Ranger*, nel quale un mezzo a sei ruote doveva attraversare una lunga serie di scenari lunari disseminati di pericolosi ostacoli. Come in *Moon Ranger*, la jeep di *Silkworm* è capace di compiere alti

balzi per superare in volata mine, rocce e altri pericoli. Questa possibilità risulta utile anche durante i numerosi attacchi aerei, nel corso dei quali vengono sganciate mine semoventi e una quantità spropositata di proiettili aria-terra.

Il procedere del fuoristrada viene ostacolato anche da una gran quantità di micidiali postazioni corazzate fisse, le cui dimensioni sono tali da rendere inutile ogni tentativo di sfuggire con un salto: in questa categoria di avversari troviamo cupole metalliche, lanciamissili a corta gittata, bocche da fuoco multiple, carri armati (che vengono paracadutati da aerei invisibili) e strani mezzi giganteschi di aspetto umanoide, dotati di un gran numero di missili.

Però è l'elicottero, stranamente simile a quell'AH-64 A di cui ci occupiamo nella recensione di *Gunship*, quello che deve affrontare il maggior numero di avversari. Elicotteri di ogni forma e dimensione, jet, astronavi globulari, proiettili vaganti e gli stessi missili lanciati dalle postazioni terrestri riempiono letteralmente lo schermo, e devono essere ridotti in frantumi utilizzando i due cannoni frontali del mezzo, di cui uno è inclinato a 45 gradi verso il suolo.

In una simile situazione, è evidente che l'arma più efficace è la collaborazione fra i due mezzi, ma fortunatamente esistono una o due peculiarità degli avversari che possiamo volgere a nostro vantaggio. Ci riferiamo per esempio alle mine terrestri, che risultano fatali al minimo contatto... ma se i giocatori riescono a bersagliare una di queste difese con un sufficiente numero di colpi, la mina esploderà rilasciando uno "shrapnel" di frammenti luminescenti che possono essere utilizzati in due modi. Il primo è quello di entrare in contatto con i frammenti, che si comporteranno come uno scudo temporaneo e impenetrabile a difesa del nostro mezzo. Sparando invece ripetutamente contro ciò che resta della mina dopo l'esplosione, questa detonerà ancora una volta cancellando tutte le forze nemiche presenti sullo schermo.

Un altro avversario che possiamo sfruttare a sua insaputa come alleato è uno strano veicolo volante di chiara ispirazione "transformer", composto da moltissimi moduli di dimensioni minori. Eliminando a colpi di cannone questo coriaceo veicolo si riescono a conquistare infatti una serie di bonus particolarmente utili. Il primo raddoppia il nostro volume di fuoco; il secondo ci dà la possibilità di rilassare lo stanchissimo dito "di sparo" fornendoci il fuoco automatico, mentre dal terzo in poi l'unico vantaggio che si ottiene è l'incremento del punteggio.

Come in ogni shoot'em up che si rispetti, il percorso è suddiviso in livelli al cui ingresso incontriamo alternativamente due tipi di "guardiani" armati molto più pesantemente del normale: si tratta di un elicottero a doppio rotore sovrapposto e di un gigantesco carro armato, che dobbiamo distruggere perché sia attivata l'assegnazione dei bonus ottenuti in precedenza.

La caratteristica principale di *Silkworm* è sicuramente la frenesia dell'azione, che non ha nulla da invidiare a quella del coin op originale. La difficoltà di molte sezioni rasenta l'impossibile, affrontandole con un solo veicolo, ragion per cui il programma rappresenta un ottimo acquisto per chi apprezza l'idea di giocare in compagnia.

Unica nota negativa a carico di *Silkworm* (che, pur essendo estremamente vario, viene caricato in una sola sezione), è l'assenza totale di un benché minimo accompagnamento musicale. La colonna sonora è interamente composta da effetti di alta qualità, che alla lunga però possono diventare piuttosto monotoni.

Complessivamente si tratta di una buona conversione che non propone idee particolarmente innovative, ma merita un giudizio senz'altro positivo per la qualità della realizzazione.

F.R.

FORGOTTEN WORLDS

Uno shoot'em up dalla trama sconclusionata e quasi ingiocabile

Computer: Amiga
Supporto: Disco
Prezzo: L. 25.000
Produzione: Capcom
Distribuzione: Leader (Via Mazzini 15, 21020 Casciago – 0332/212255)

Alcune trame sono decisamente poco serie. Certo, nessuno pretende una logica ineccepibile nell'ambientazione di uno shoot'em up, ma lo scenario di *Forgotten Worlds* è eccezionalmente assurdo.

Stando a quel che racconta il manuale, una divinità demoniaca è sorta all'improvviso per riorganizzare il mondo secondo i suoi gusti personali. La prima mossa del cornuto personaggio è stata quella di radere al suolo più palazzi possibile, dopodiché l'amministrazione della Terra, divisa in quattro regni, è stata affidata a mostruosi e colossali consiglieri, ognuno a capo di uno spaventoso esercito biomeccanico.

È superfluo dire che contro avversari tanto potenti nessuno pensa neanche lontanamente di sferrare un contrattacco, ma dal momento che i progettisti del gioco avevano assolutamente bisogno di qualcuno da lanciare attraverso le linee nemiche, ecco farsi avanti una coppia di eroi...

I protagonisti di *Forgotten Worlds* sono due omoni, i quali, dopo aver indossato un completo adatto all'occasione composto da canottiera e calzamaglia (la distruzione ha probabilmente raso al suolo anche tutti i negozi d'abbigliamento!), partono con sommo sprezzo del pericolo verso il quartier generale del primo Guardiano. Per aiutarsi nell'impresa entrambi imbracciano uno strano marchingegno capace di sollevarli nell'aria, e si trovano così a poter svolazzare su tutta l'area dello schermo.

Sin dall'inizio del gioco, accompagnato da una musica funerea e lentissima, gli eserciti demoniaci si accaniscono con tutte le forze disponibili contro i giocatori: truppe volanti e terrestri, postazioni d'artiglieria, droidi lanciamissili, semoventi a forma di totano e decine di altri orrori affollano lo schermo con i loro proiettili che riducono la nostra energia vitale. Mentre lo scroll orizzontale procede implacabile avvicinandoci alla tana del Guardiano, possiamo cercare di raccogliere i gettoni metallici che appaiono in seguito alle esplosioni dei nostri avversari: questi gettoni, chiamati "zenny", vengono usati come denaro nei negozi che appaiono dal suolo nei momenti più inaspettati.

I negozi, gestiti inspiegabilmente da stupende fanciulle per nulla turbate dagli avvenimenti esterni, offrono una quantità di servizi utili. Avendo a disposizione un'adeguata quantità di zenny, è possibile acquistare armi di ogni foggia e dimensione, curare le ferite, proteggerci con armature od ottenere informazioni su ciò che ci attende nei quadri successivi.

Indipendentemente dall'incoerenza della trama, della presentazione e dello svolgimento del gioco, *Forgotten Worlds* è un'ottima conversione del poco conosciuto e recentissimo coin op omonimo. Gli scenari, tetri al punto giusto, riproducono con fedeltà assoluta quelli originali, e uniti alla fluidità dell'animazione fanno apprezzare in pieno le capacità grafiche dell'Amiga, sfruttate quasi ai livelli massimi.

L'azione, prevista per due giocatori, rende praticamente impossibili le partite singole. Il difetto più grande di *Forgotten Worlds*, infatti, sta nell'estrema difficoltà di controllo che spesso conduce a decessi quantomai immotivati e spiacevoli. Questo aspetto influisce pesantemente sul giudizio complessivo, visto che il programma (nonostante il suo impressionante aspetto audiovisivo), risulta quasi totalmente ingiocabile.

Un'ottima conversione... nella quale probabilmente nessuno riuscirà a vedere mai più del primo livello.

F.R.

GUNSHIP

Un buon gioco che vede come protagonista l'elicottero Apache AH-64 A

Computer: Amiga
Supporto: Disco
Prezzo: L. 59.000
Produzione: Microprose
Distribuzione: Lago (Via Napoleona 16, 22100 Como – 031/300174)

Il mezzo simulato in *Gunship* è l'elicottero americano AH-64 A, meglio conosciuto come "Apache": un mezzo considerato unanimemente come uno dei migliori velivoli militari multiruolo. Sistemi di puntamento controllati dallo sguardo del pilota, avanzati sistemi di guerriglia elettronica, pale tanto resistenti da poter tagliare le cime degli alberi: questi sono solo alcuni degli aspetti più spettacolari dell'Apache, il cui prezzo (circa sette milioni di dollari) permette addirittura di classificarlo tra le armi economiche!

Rispetto alla già acclamata versione per Commodore 64, la versione per Amiga di *Gunship* offre molte novità, a partire dalla maggior velocità di risposta dei comandi. Gli altri miglioramenti riguardano com'è ovvio la grafica – che non raggiunge comunque livelli particolarmente elevati di spettacolarità – il sonoro e le opzioni di gioco. È possibile volare con diversi livelli di realismo, in missioni di pace o di guerra, che in questa nuova versione replicano eventi reali avvenuti dal 1960 a oggi.

Il giocatore può scegliere il luogo del conflitto, l'armamento in dotazione all'Apache e il fattore di realismo ambientale, che può arrivare a simulare i venti a varie quote e la temperatura dell'aria. Dopo queste selezioni preliminari, tutto quello che resta da fare è scegliere il proprio nome di battaglia, comunicare la propria disponibilità al Comando e attendere qualche istante per essere inviati verso la nostra prima missione.

Una volta a bordo, con un certo stupore si nota subito la relativa scarsità di strumenti: chi è abituato alle decine d'indicatori dei tradizionali simulatori di volo può pensare che i progettisti abbiano dimenticato qualche strumento, ma a tutto c'è una spiegazione. Lo AH-64 A è un veicolo in cui l'elettronica gioca un ruolo fondamentale: molte funzioni secondarie sono affidate ai computer di bordo, e il pilota deve tenere d'occhio soltanto tre strumenti durante le manovre, ed è quindi relativamente libero di concentrarsi sulla missione, affrontando azioni altrimenti impossibili.

Osservando la mascherina che va sovrapposta alla tastiera del computer, ci si rende conto che la maggior parte dei comandi non riguardano la navigazione ma il combattimento: tasti contraddistinti da sigle un po' oscure controllano la selezione, il puntamento e il lancio delle quattro armi di bordo, mentre un secondo set di comandi serve per le contromisure elettroniche. Siamo quasi confusi dalla quantità di comandi, ma ci viene in aiuto un prezioso alleato, il TADS. In questo acronimo si cela l'avanzatissimo sistema di puntamento automatico del veicolo, che in nostra vece seleziona i bersagli, identificandoli in modo autonomo e avvertendoci non appena entrano nel raggio d'azione delle nostre armi.

La missione-tipo di *Gunship* fa un ampio uso sia del TADS sia del sistema di navigazione automatico comandato dalla mappa che possiamo richiamare con un apposito tasto.

Secondo una procedura già nota ai piloti di jet, ma fino a oggi inesistente sui velivoli ad ala mobile, il pilota può selezionare con un cursore la propria destinazione, affidando a un indicatore elettronico il compito di tenere sotto controllo la rotta. Tutte queste caratteristiche hanno lo scopo di automatizzare alcune delle funzioni più complesse, permettendo a una persona sola di compiere missioni anche molto elaborate, che in genere richiedo-

no il raggiungimento di più obiettivi indipendenti.

Come nella maggior parte dei giochi di simulazione siglati MicroProse, l'esito di ogni missione viene registrato permanentemente su disco andando a formare un registro della carriera militare del giocatore. Sulla base di questi dati, il computer (o forse sarebbe meglio dire "il Comando") decide le promozioni, le decorazioni e le eventuali degradazioni, che vengono rappresentate con adeguate schermate statiche. Questo aspetto dell'ambiente simulato, unito alla soddisfazione che dà una sempre maggiore padronanza del veicolo, aiutano ad aumentare la longevità di un programma dotato già di per sé di notevole fascino.

Gunship, con il suo manuale di 83 pagine che tratta molti aspetti del pilotaggio e del combattimento, con la sua grafica veloce e funzionale e il suo realismo d'ambientazione, va sicuramente considerato un buon gioco di simulazione. La manualistica, curata, unita alla semplicità di gestione del mezzo, permette sin dalle prime partite di padroneggiare discretamente il velivolo, e questa caratteristica fa di *Gunship* un prodotto piacevole anche per chi desidera un "normale" gioco di guerra e non sia patito esclusivamente di simulazioni di volo. Sia chiaro comunque che solo una grande pratica permette di ottenere prestazioni davvero eccellenti.

Va comunque fatta una precisazione: *Gunship* non è un simulatore di volo che tenta di riprodurre con elevatissimo realismo ogni particolare della navigazione. Gli appassionati del genere, per intenderci i "piloti" di *Flight Simulator II* e *Falcon*, potrebbero rimanerne delusi. Da questo punto di vista, infatti, riteniamo più corretto definire *Gunship*, un gioco di simulazione... di ottimo livello, ma pur sempre un gioco. **F.R.**

STAR TREK

«E questi saremmo noi?» chiese il signor Spock con aria impassibile

Computer: C-64/128
Supporto: Cassetta/Disco
Prezzo: L. 18.000/25.000
Produzione: Firebird
Disponibile presso: Lago (Via Napoleona 16, 22100 Como – 031/300174)

Penso che ben pochi non abbiano visto o sentito parlare del capitano Kirk, del signor Spock e delle loro avventure in galassie inesplorate, alla ricerca di nuovi mondi e nuove civiltà. Oltre alla saga televisiva, che nonostante l'età riesce ancora ad appassionare giovani e meno giovani, gli eroici personaggi di Star Trek hanno ispirato produzioni di ogni tipo: gadget, riviste, film, libri, modellini... non potevano certo mancare i videogiochi.

Una missione all'insegna della tecnologia, della scienza e della giustizia dovrebbe essere la trama ideale per ispirare un videogioco ricco di strategia, d'azione e di tutti quegli elementi futuristici e futuribili tanto cari ai programmatori... Invece si è persa l'occasione di sfruttare fino in fondo il complesso background costruito in oltre vent'anni dai soggettisti di film e telefilm, e si è preferito puntare su un giochetto elementare e poco soddisfacente, anche se con qualche spunto simpatico.

Il programma di cui ci occupiamo in questa recensione, infatti, ha ben poche delle caratteristiche che hanno reso celebre la saga di Star Trek: si tratta di un gioco statico, dove l'azione è latitante e la strategia praticamente inesistente. Probabilmente anche gli appassionati che non si sono persi una sola puntata delle avventure di Spock e compagni resteranno delusi, nonostante la riproduzione della cabina di comando dell'ormai mitica *USS Enterprise* con l'equipaggio al completo, e la sigla iniziale piuttosto fedele a quella della serie televisiva.

In realtà coloro che già da qualche tempo seguono l'evoluzione delle varie edizioni di giochi ispirati a questa saga, troveranno che almeno dal punto di vista grafico e sonoro il prodotto della Firebird è decisamente superiore ai suoi predecessori, ma nonostante ciò sembra proprio che questo tema non riesca in alcun modo a ispirare situazioni in cui tattica e azione abbiano una qualche rilevanza.

Esaminiamo lo svolgimento del gioco. L'*Enterprise* si trova all'interno di una galassia ribelle e si muove da un sistema solare all'altro con il compito di riportare la situazione alla normalità. Per completare la missione viaggia lungo le rotte stellari, abbattendo le navi nemiche che si frappongono tra lei e il suo obiettivo e inviando i propri uomini sulla superficie dei vari pianeti.

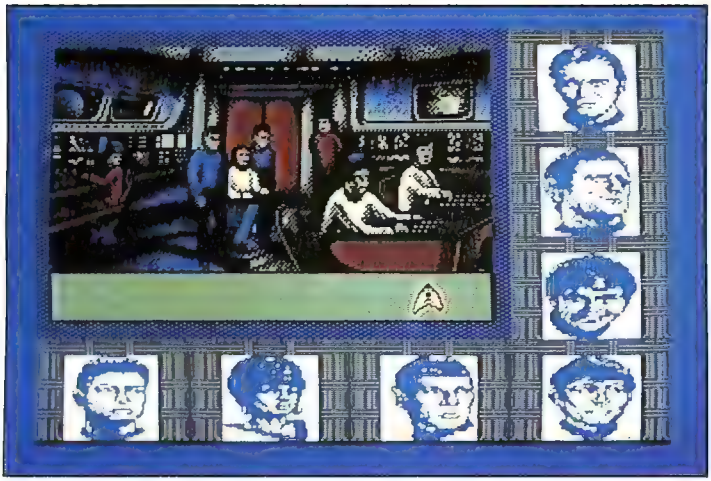
Il metodo usato per impartire gli ordini è piuttosto interessante: lo schermo di gioco è occupato al centro dall'immagine del ponte di comando della nave con tutti i membri dell'equipaggio, mentre sul bordo compaiono i volti dei famosi ufficiali dell'*Enterprise*. Per ordinare un'operazione è sufficiente portare il cursore sulla casella dell'ufficiale responsabile (per esempio Sulu per stabilire la rotta, il tenente Uhura per le comunicazioni e così via).

Ogni volta che si attiva una casella, compaiono uno o più sottomenu che servono per individuare chiaramente l'operazione da compiere. Ma tutti questi sforzi per aiutare il giocatore appaiono superflui, considerata la staticità e la piattezza del gioco. Paradossalmente, avremmo preferito trovare qualche difficoltà almeno nella fase di "familiarizzazione" con il programma, visto che in seguito le cose diventano

ancora più facili. Le uniche azioni intraprese dal giocatore si riducono agli spostamenti ordinati da Sulu e alle operazioni di mira e di fuoco durante gli attacchi delle inevitabili astronavi dei Klingon. La generale lentezza rende piuttosto semplice colpire il nemico con tutta la potenza delle proprie armi (phaser e siluri fotonici) e di sgombrare la strada al cammino dell'astronave pacificatrice.

Le cose non migliorano nemmeno quando è il momento di scendere sui pianeti, visto che l'azione del giocatore si limita alla scelta tra una serie di proposte e non ci vuole molto a scoprire qual è il giusto modo d'impostare le proprie decisioni. Dopo aver deciso la composizione del gruppo esplorativo, i nostri eroi si trovano di fronte a una serie di problemi da risolvere: ogni componente espone il suo parere e il giocatore non deve fare altro che decidere quale proposta rappresenta la via migliore per superare l'ostacolo.

Non tutto il programma è però un concentrato di elementi negativi, bisogna anche dire che esistono alcune schermate grafiche di tutto rispetto (per esempio quella che rappresenta tutti i sistemi della galassia in movimento e che appare al momento della scelta della rotta) e, nonostante siano piuttosto meccaniche in quanto "obbligate" e limitate, le possibilità di scelta del programma costituiscono un buon numero di varianti del gioco. I problemi nascono quando si cominciano a ritrovare troppo spesso opzioni già viste, cosa che peraltro non accade dopo moltissimo tempo; a quel punto l'avventura diventa sempre più ripetitiva e in capo a poche ore il gioco perde decisamente di fascino e d'interesse.



In definitiva, se vi rende felici vedere i vostri eroi ben rappresentati sullo schermo del monitor, l'acquisto di questo gioco potrebbe rivelarsi un buon investimento, ma se amate Star Trek perché vi piacciono l'emozione e l'avventura, è molto improbabile che riusciate a divertirvi vedendo i vostri eroi impegnati in una missione in cui la partecipazione è ridotta al minimo. Gli stessi ufficiali dell'*Enterprise* digitale non sembrano particolarmente soddisfatti della loro sorte!

N.F.R.

3D POOL

Il più spettacolare biliardo a 8 bit disponibile sul mercato

Computer: C-64/128
Supporto: Cassetta/Disco
Prezzo: L. 18.000/25.000
Produzione: Firebird
Distribuzione: Leader (Via Mazzini 15, 21020 Casciago – 0332/212255)

Spesso l'idea del computer "intelligente" viene associata alla capacità da parte della macchina di giocare a scacchi, ma esiste un secondo gioco che rappresenta una sfida di programmazione ancora maggiore. Stiamo parlando del biliardo, passatempo considerato forse poco nobile per l'atmosfera di fumo e bassifondi che circonda il tavolo verde nell'iconografia classica, ma in realtà intellettualmente impegnativo e senz'altro divertente. Non dimentichiamo che il biliardo risale nelle sue versioni più primitive al XVI secolo e che fu il divertimento prediletto del Re Sole, Luigi XIV. E nel 1835 un matematico francese, de Coriolis, pubblicò addirittura un trattato di meccanica dedicato a questo gioco.

Probabilmente i programmatori di oggi non conoscono il libro di Coriolis, ma è certo che dietro a ogni biliardo elettronico si celano studi attenti e particolareggiati.

Durante i primi giorni del Commodore 64, quando il computer disegnato da Porsche era ancora troppo costoso per molte tasche, gli ingegneri dei mitici HAL Laboratories scrissero un programma per molti versi imbattuto. Supportato da una cartridge, il *Billiard 64* rendeva possibile una gran quantità di giochi, ognuno dei quali sfruttava in maniera magistrale le capacità grafiche e di calcolo della macchina.

Praticamente sino a oggi quel-

l'introvabile programma costituiva l'unica scelta possibile per chi volesse esplorare seriamente i misteri del tavolo verde con il proprio Commodore: gli analoghi titoli pubblicati nel corso degli anni (come il terribile *Steve Davis' Snooker*) facevano solo rimpiangere quella cartuccia. Ci pensa ora la Firebird a portare nuova linfa a questo particolare campo del divertimento elettronico.

3D Pool, che esiste anche in versione per l'Amiga, è graficamente il biliardo elettronico più spettacolare che esista (almeno per quanto riguarda i programmi per home computer). La sua caratteristica più coinvolgente è infatti la grafica, che utilizza routine per la visualizzazione di solidi non lontane da quelle denominate Freescape. Durante il gioco è possibile cambiare il proprio punto di vista del tavolo avvicinandosi sino a pochi millimetri dalla palla o osservando il panno verde da metri e metri di distanza. Questa notevole possibilità, che viene gestita interamente da tastiera, non è affatto un optional superfluo: dal momento che non viene visualizzata nessuna stecca, è proprio la linea che unisce il centro dello schermo alla palla che "spinge" la sfera, e la nostra visione della biglia determina la forza e l'inclinazione del tiro.

Confusi? È una sensazione comune a tutti quando ci si avvicina a un programma tanto innovativo, ma viene superata dopo pochi minuti passati davanti allo schermo. Una volta che ci si è divertiti a ruotare in tempo quasi reale il tavolo in tutte le direzioni, comunque, comincia il vero divertimento. Tra le decine di giochi che hanno come base il tavolo verde e la stecca, quello messo a disposizione da *3D Pool* è tra i più famosi. Basterà un sommario accenno alle regole e quasi tutti lo riconosceranno immediatamente (forse un piccolo limite del programma è proprio l'impossibilità di cimentarsi in altri tipi di partita).

All'inizio del gioco, sul tavolo sono disposte in triangolo 15 biglie numerate: quelle indicate con le cifre da 1 a 7 e di colore uniforme appartengono a un giocatore, quelle numerate da 9 a 15 e che sfoggiano una striscia bianca appartengono al secondo giocatore. La biglia numero 8 è completamente nera, e ha una funzione particolare. La partita, che come avrete intuito è una sfida a due, vede entrambi i contendenti usare una sedicesima biglia bianca per raggiungere l'obiettivo d'imbucare tutte le proprie sfere e concludere la serie mandando in un'apposita buca la biglia nera.

Fra il dire e il fare in questo caso non c'è solo il mare, ma soprattutto le biglie dell'avversario, che bisogna cercare di evitare o di spostare in punti per lui svantaggiosi. Durante la partita si scoprono un gran numero di regole minori, che si riferiscono soprattutto all'uso scorretto della biglia nera.

I giocatori controllati dal computer possono essere uno o persino tutti e due, ma se giocato in solitario *3D Pool* offre un'ulteriore sfida, rappresentata dai campionati, selezionabili tramite un'apposita opzione. Partendo dai quarti di finale lo scopo è di sfidare e battere il mitico Maltese Joe, un vero campione di biliardo che ha avuto il ruolo di consulente nella realizzazione del programma.

L'impresa non è per nulla facile, e forse anche in funzione di allenamento esiste la possibilità di particolari situazioni denominate "trick shot" nelle quali ci viene richiesto di mandare in buca con un solo colpo tutte le biglie presenti sul tavolo. Ogni tiro può essere studiato con un editor incorporato, utile anche per preparare i propri "trick shot".

Acquisto indispensabile per tutti gli amanti del gioco reale, consigliamo *3D Pool* anche a coloro che si ritengono in grado di apprezzare un gioco intelligente e ben presentato. Avvertiamo però che il biliardo elettronico potrebbe risultare ben presto troppo complicato e monotono per... i comuni mortali.

F.R.

SOFTWARE HELPLINE

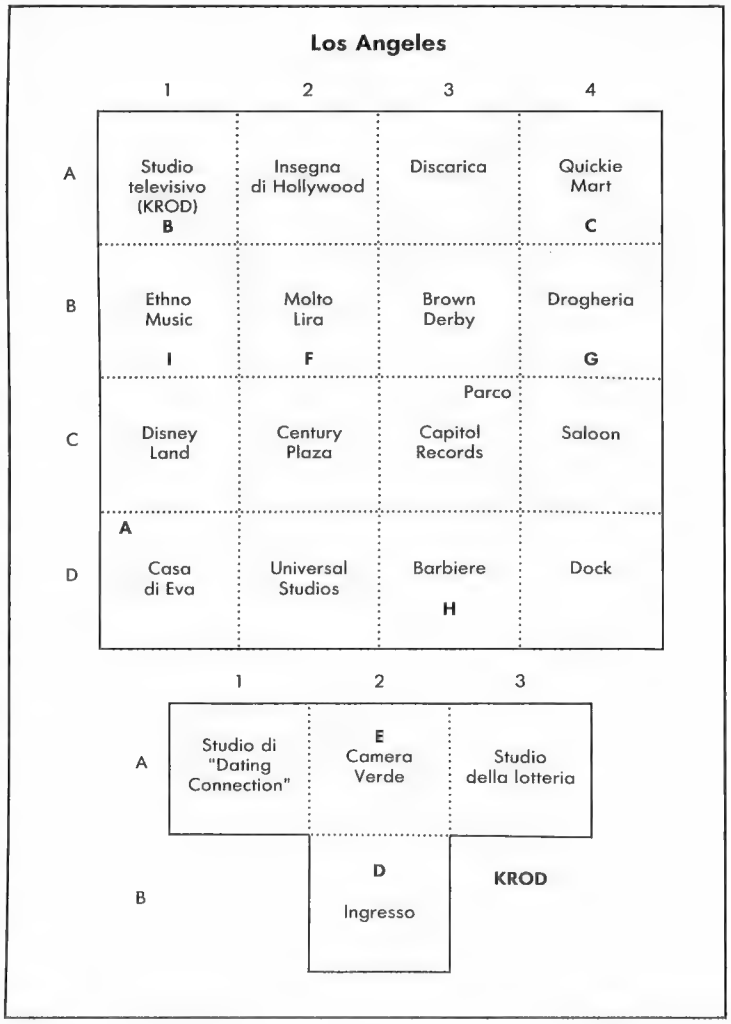
Leisure Suit Larry II: la soluzione

Los Angeles: ***Casa di Eva (D1)*** – Entrate nel garage. Muovetevi verso est sinché non scomparite alla vista. Prendete la banconota. Andate alla KROD (A1).
KROD (A1) – Guardate (prendete nota dell'informazione, che vi sarà utile in seguito). Andate nella discarica (A3).
Discarica (A3) – Camminate sino al buco nella palizzata. Guardate nel buco. Andate da Quickie Mart (A4).
Quickie Mart (A4) – Parlate alla ragazza. Comprate un biglietto della lotteria. Inserite uno qualsiasi dei sei numeri di tre cifre. Andate alla KROD (A1).

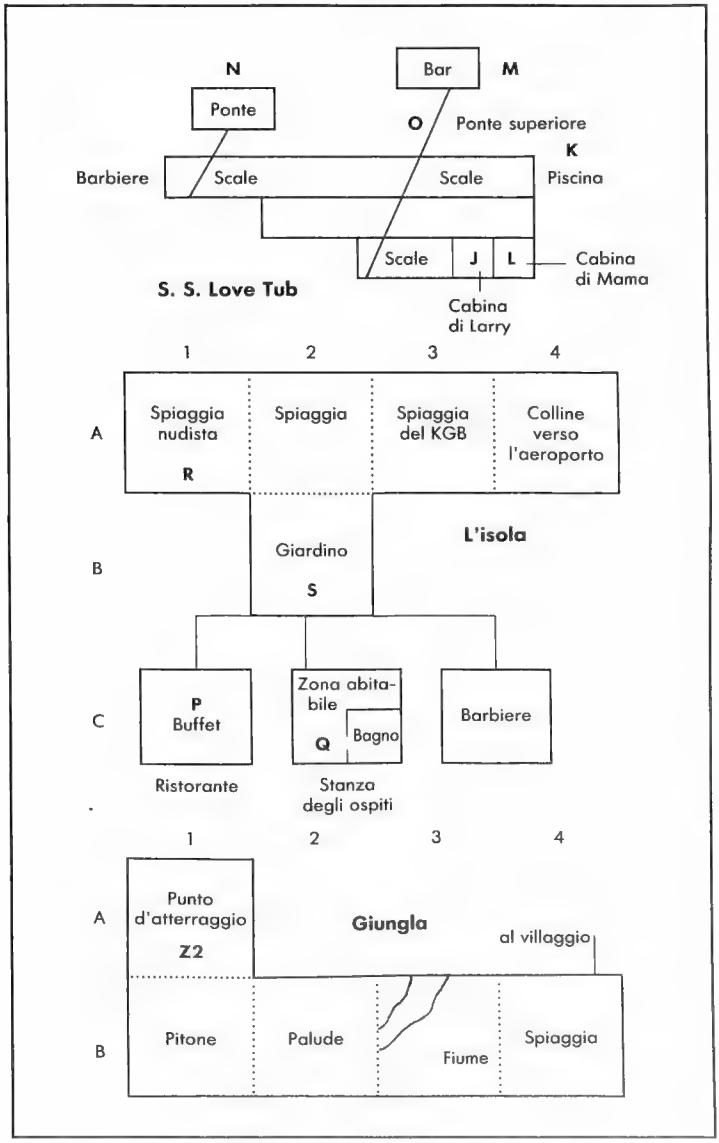
KROD: ***KROD (A1)*** – Entrate nel palazzo.
Ingresso (B2) – Mostrate il biglietto alla ragazza. Scrivetevi il numero che vi viene dato dalla ragazza alla reception (la ragazza vi chiederà il numero). Ripetete il numero datovi alla reception. Entrate nella Camera Verde tramite la porta a nord.
Camera Verde (A2) – Sedetevi sulla panca (entra un uomo). Seguite l'uomo attraverso la porta ovest.
The Dating Connection – Date una qualsiasi risposta alle domande (vincete il gioco). Tornate alla Camera Verde.
Camera Verde – (L'uomo vi dà il biglietto per la crociera). Sedetevi sulla panca (entra una donna). Seguite la donna attraverso la porta a est.
Studio della lotteria – Vincete un milione di dollari per ogni anno della vostra vita, prendete il denaro dalla ragazza e tornate indietro nell'ingresso.
Ingresso – Lasciate il palazzo. Andate a casa di Eva.

Los Angeles: ***Casa di Eva*** – Guardate nella spazzatura (due volte). Prendete il passaporto. Andate a Century Plaza (C2).
Century Plaza (C2) – Guardate l'uomo. Andate a Molto Lira (B2).
Molto Lira (B2) – Leggete il cartello. Prendete un costume da bagno nel retro della stanza. Comprate il costume davanti al banco. Andate in drogheria (B4).
Drogheria (B4) – Andate all'estremità ovest del ripiano all'estrema sinistra. Guardate il ripiano. Prendete la lozione. Comprate la lozione. Andate da Quickie Mart (A4).
Quickie Mart (A4) – Andate alla macchinetta distributrice di soda. Prendete la soda. Comprate la soda. Andate dal barbiere (D3).
Barbiere (D3) – Parlate con l'uomo. Sedetevi su una sedia. Fatevi tagliare i capelli. Andate al negozio di musica (B1).
Negozio di musica (B1) – Parlate alla ragazza. Prendete il microfilm (appare il sosia del KGB). Andate al dock (D4).
Dock (D4) – Date il biglietto all'uomo. (Nota: Ponte F, Cabina 1). Salite la passerella e andate sulla nave.

S.S. Love Tub: ***Ponte inferiore*** – Andate a est per entrare nella cabina di Larry.
Cabina di Larry – Prendete la frutta. Leggete la nota. Andate nella parte est della camera. Aprite la porta. Andate a est nella cabina di Mama.
Cabina di Mama – Non avvicinatevi troppo a meno che non abbiate salvato il gioco e vogliate farvi una risata. Ascoltate il discorso di Mama. Andate a ovest nella cabina di Larry.
Cabina di Larry – Chiudete la porta. Indossate il costume. Andate a sud per lasciare la stanza.
Ponte inferiore – Andate a ovest sulle scale, poi su per due livelli.
Ponte superiore – Lasciate le scale e camminate verso est per arrivare nella zona della piscina.
Zona piscina – Andate dentro alla piscina. Nuotate. Immergetevi. Nuotate sin sul fondo della piscina. Prendete il top. Nuotate in superficie. Lasciate la piscina. Andate verso la sdraio vuota. Usate la lozione. Sedetevi. (Una ragazza vi parla: ignoratela). Alzatevi. Tornate alla cabina di Larry.
Cabina di Larry – Salvate il gioco. Aprite la porta sulla parete est. Entrate nella cabina di Mama.
Cabina di Mama – Aprite il comodino. Guardate nel comodino. Prendete il kit. Se arriva Mama, ricaricate e ritentate. Lasciate



la stanza.
Camera di Larry – Mettetevi i vestiti. Andatevene e non tornate; dopo qualche ingresso Mama vi acchiappa. Tornate al ponte superiore e andate tutto a ovest.
Barbiere – Sedetevi sulla sedia. Prendete la parrucca. Camminate a est, poi in alto sino alle scale verso est.
Bar – Andate all'estremità occidentale del bar. Guardate. Non ordinate nulla. Lasciate il bar.
Ponte superiore – Andate sulle scale a ovest e salite sul ponte.
Ponte – Andate nella parte orientale della camera. Muovete le maniglie (questo fa partire l'argano della scialuppa di salvataggio). Tornate indietro sulle scale a est.
Ponte superiore – Camminate sino alla linea verde (a metà fra il ponte superiore e il bar) e andate a ovest verso le scialuppe. Entrate nella scialuppa.
Sulla scialuppa – La scialuppa parte, e dovete fare le due seguenti azioni prima che la scena cambi: indossate la parrucca. Mangiate lo snack. Dopo che la scena è cambiata, pescherete con il kit da cucito, poi raggiungerete un'isola. Nuotate sino alla costa.



L'isola: ***Spiaggia di arrivo (A2)*** – Andate a ovest sino alla spiaggia dei nudisti.
Spiaggia nudista (A1) – Parlate alla donna (non seguite la ragazza). Camminate verso est sino al punto di arrivo (dopo una breve attesa).
Spiaggia di arrivo (A2)– Camminate verso sud.
Giardino (B2) – (Vagate nel giardino).
Ristorante (C1) – Parlate all'uomo. Pagatelo. Sedetevi. Camminate sino al bar. Guardate il cibo. Prendete il coltello (non mangiate il cibo). Lasciate il ristorante.
Giardino (B2) – (Vagate in giro).
Stanza degli ospiti (C2) – Non seguite la cameriera. Guardate nel comodino. Prendete i fiammiferi. Andate in bagno. Prendete il sapone. Lasciate la stanza.
Giardino (B2) – (Vagate in giro).
Barbiere (C3) – Sedetevi (vi ritrovate biondi). Lasciate il negozio.
Giardino (B2) – (Vagate un altro po').
Spiaggia di arrivo (A2) – Andate a ovest alla spiaggia dei nudisti.
Spiaggia nudista (A1) – Guardate le rocce. Prendete le mutandine del bikini. Andate a est sulla spiaggia di arrivo.
Spiaggia di arrivo (A2) – Andate a sud nel giardino (ripetete i passi necessari per arrivare alla stanza degli ospiti).
Stanza degli ospiti (C2) – Andate nell'area abitabile oltre il bagno. Indossate il bikini. Riempite il top con il sapone. Tornate nel giardino e andate dal barbiere.
Barbiere (C3) – Sedetevi. (Vi fanno una ceretta). Tornate in giardino.
Giardino (B2) – Guardate i fiori. Prendete un fiore (attendete sinché non siete vicini alla palma in centro per raccoglierlo).
Spiaggia di arrivo (A2) – Andate a est sino alla spiaggia del KGB.
Spiaggia del KGB (A3) – Superate gli agenti del KGB.
Colline (A4) – Attraversate le colline. (Ottenete un punto extra per ogni caduta ma li perdete alla fine del gioco). Vedete l'aeroporto. Cambiatevi d'abito prima che la scena cambi.

Aeroporto: ***Ingresso del terminal (D2)*** – Andate dall'Hare Krishna. Date il fiore all'uomo.
Bancone (C2) – Andate a est sino all'area di attesa.
Area d'attesa (C1) – Andate a nord dal barbiere.
Barbiere (B1) – Sedetevi (attendete il taglio di capelli). Tornate al bancone e andate ad est. Mostrate il passaporto.
Customs (C3) – Camminate verso gli addetti. Mostrate il passaporto e andate a est attraverso la porticina.
Zona di sicurezza (C4) – Andate al nastro trasportatore. Prendete la borsa mentre passa (ripetete sino a quando non prendete la borsa con la bomba e tornate al bancone

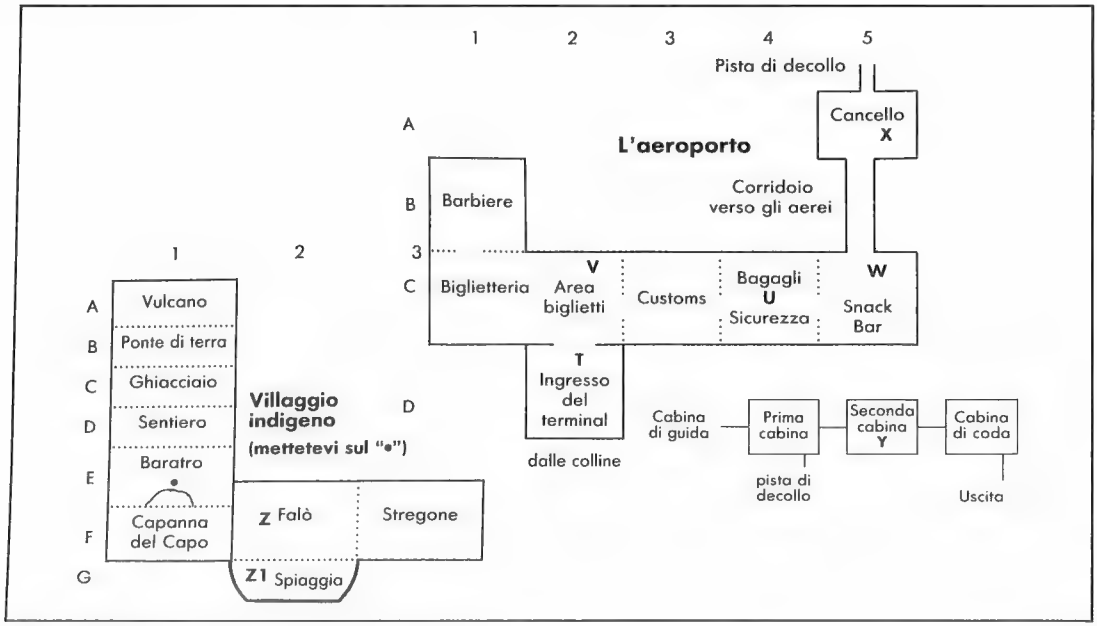
della biglietteria).
Biglietteria (C2) – Mettetevi in fila (dopo che la bomba è esplosa). Parlate alla ragazza (ottenete un biglietto). Andate a est. Mostrate il passaporto. Andate nella zona di sicurezza.
Customs (C3) – Mostrate il passaporto e andate a est verso la zona di sicurezza.
Zona di sicurezza (C4) – Andate a est sino allo snack bar.
Snack bar (C5) – Leggete l'insegna. Comprate lo special. Guardate nel cibo. Prendete lo spillo. Andate al distributore automatico superiore. Guardate la macchina. Comprate un'assicurazione. (Ricevete un paracadute). Andate nel corridoio.
Cancello (A5) – Guardate il bancone. Prendete il pamphlet. Date il biglietto all'uomo (andate oltre la porta a nord e nell'aereo).
Sezione anteriore (A1) – Date il biglietto alla donna e camminate nella sezione mediana.
Sezione mediana (A2) – Guardate il sedile. Prendete la borsa. Alzatevi. Date il pamphlet all'uomo (dopo che vi ha parlato.) Alzatevi. Andate nella sezione posteriore.
Sezione posteriore (A3) – Andate in coda all'aereo. Guardate le porte. Andate nella parte sud dell'aereo. (C'è una porta ma non potete vederla). Indossate il paracadute. Scassinate la serratura. Girate la maniglia (venite strappati dall'aereo). Tirate la corda del paracadute (velocemente!). (Cadete su un albero). Usate il coltello.

Giungla: ***Punto di atterraggio (A1)*** Guardate al suolo. Prendete il bastone. Andate a sudovest, subito a nord del cespuglio. (Evitate l'albero: contiene api assassine.) Strisciate. Andate a sud.
Pitone (B1) – Andate a sud (sino a che il serpente non comincia a muoversi). Usate il bastone (velocemente!). Andate a est nella palude.
Palude (B2) – (Salvate il gioco). Andate a est lungo il tragitto giallo chiaro.
Fiume (B3) – Andate sulla riva del fiume e dondolatevi su una liana (poiché dovete farlo velocemente per tre volte, conviene usare il tasto F3). Lasciate la liana. Prendete la liana. Andate a est sulla spiaggia (la ragazza vi porta dal capo del villaggio).

Villaggio indigeno: ***Capanna del capo (F1)*** e ***Baratro (E1)*** (Seguite il capo a nord verso il baratro. Ascoltatelo). Andate a sud sino alla capanna del capo, poi a est dal falò.
Falò (F2) – Prendete le ceneri. Andate a sud sulla spiaggia.
Spiaggia (G2) – Prendete la sabbia. Tornate nella zona del crepaccio.
Baratro (E1) – (Salvate il gioco). Guardate l'albero. Andate a nord sino alla fine centrale del sentiero. Lanciate la liana. (Se non riuscite, tentate da un altro punto). Andate a nord (dopo aver attraversato il baratro).
Ghiacciaio (C1) – Gettate la sabbia (o le ceneri) sul ghiaccio. Andate a nord sul ponte di terra.
Ponte di terra (B1) – Andate a nord oltre il ponte.
Vulcano (A1) – Andate verso il centro dello schermo presso il crepaccio. Aprite il tonico. (Dovete essere nel punto giusto, indicato anche sulla mappa.) Mettete il sacchetto per i malesseri aerei nel tonico. Accendete il sacchetto con i fiammiferi. Gettate il tonico nel crepaccio. Entrate nell'ascensore.

LEGENDA DELLA MAPPA

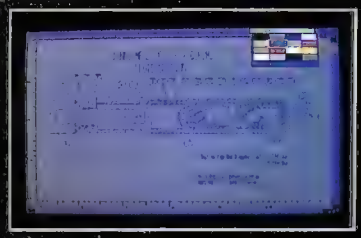
A: Banconota (garage), Passaporto (fra la spazzatura).
B: Nota.
C: Biglietto della lotteria, soda.
D: Numeri vincenti (ragazza).
E: Biglietto per crociera (uomo).
F: Costume da bagno.
G: Lozione.
H: Taglio di capelli, parrucca, ceretta, taglio, tonico.
I: Microfilm.
J: Frutta, nota (cabina di Larry).
K: Top del bikini (sul fondo della piscina).
L: Kit da cucito (comodino).
M: Snack.
N: Maniglie dell'argano di servizio.
O: Scialuppa di salvataggio.
P: Cibo, coltello.
Q: Fiammiferi (comodino), sapone (bagno).
R: Mutandine del bikini.
S: Fiore.
T: Hare Krishna.
U: Borsa con bomba (sul nastro trasportatore).
V: Biglietto.
W: Spillo (nel cibo), paracadute (macchina distributrice superiore).
X: Pamphlet
Y: Sacco (sedile).
Z: Ceneri.
Z1: Sabbia.
Z2: Bastone. ■



I PRIMI 10

programmi scelti dai lettori

1.	TV Sports Football	(Amiga)
2.	Starglider 2	(Amiga)
3.	Pool of Radiance	(C-64)
4.	R-Type	(Amiga)
5.	R-Type	(C-64)
6.	Dragon's Lair	(Amiga)
7.	TrianGO	(C-64)
8.	MicroProse Soccer	(C-64)
9.	Falcon	(Amiga)
10.	Rocket Ranger	(C-64)



IL PROGRAMMA DEL MESE

scelto dalla redazione

Draw 2000 (Amiga)

[...] Il pacchetto è completo sotto tutti i profili: cerca sempre di sfruttare al meglio le caratteristiche dell'Amiga, traendosi d'impaccio in maniera egregia quando si scontra con i limiti hardware della macchina.

(da Commodore Gazette numero 2/-89)

MODALITÀ PER LA VOTAZIONE

Classifica Software, è un'occasione che Commodore Gazette dà ai suoi lettori per esprimere il proprio parere sui migliori programmi del momento (sia giochi che utility). Per votare, compilare (anche solo parzialmente) il tagliando riportato in fondo alla pagina, ritagliare, e spedire a:

Commodore Gazette, Classifica software
Via Monte Napoleone, 9 – 20121 Milano

NOME _____
COGNOME _____ ETÀ _____
INDIRIZZO _____

SECONDO ME, I MIGLIORI PROGRAMMI DEL MOMENTO SONO (nome e versione):

1 . _____
2 . _____
3 . _____
4 . _____
5 . _____
6 . _____
7 . _____
8 . _____
9 . _____
10. _____

COMPUTER POSSEDUTO (marca e modello):

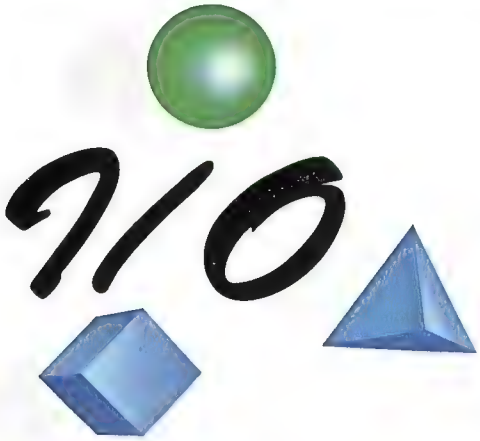
_____

LEGGETE COMMODORE GAZETTE:

☐ Abitualmente
☐ Raramente
☐ Questa è la prima volta

INPUT / OUTPUT

INPUT/OUTPUT SVELA I SEGRETI DEL VOSTRO COMPUTER



C-64/128 in modo 64

***197 La funzione USR() -** La funzione USR() viene di rado utilizzata nel C-64 in quanto per farne un uso appropriato occorre conoscere il linguaggio macchina e il kernel, e inoltre bisogna essere capaci di convertire un'espressione algebrica in una sequenza di subroutine in LM e di costanti in virgola mobile. In generale, quindi, si rinuncia a questa funzione e si continuano a usare la comoda istruzione FN del Basic quando vi è necessità di definire funzioni come per esempio l'arcoseno. Le due routine in LM che seguono vengono attivate proprio impiegando la funzione Basic USR().

La prima routine svolge un compito molto semplice, ossia fornisce il valore numerico della word che si trova all'indirizzo specificato come argomento della funzione USR(): ciò equivale a eseguire l'istruzione Basic PEEK(indirizzo)+PEEK(indirizzo+1)*256. Il programma installa la routine in memoria e tramite la SYS 16384 abilita la funzione USR(). Se per esempio desideriamo che nella variabile A finisca il contenuto del vettore di interrupt a $0314, è sufficiente inserire nel proprio programma Basic l'istruzione `A = USR(788)`. Ovviamente, l'argomento della funzione può essere una qualunque espressione algebrica che restituisca un valore numerico.

```
100 FOR I=16384 TO 16436:READA:C=C+A:POKEI,A:NEXT
110 IF C<>4354 THENPRINT"ERRORE NEI DATA...":END
115 SYS16384
120 DATA 169 64.141 18.3.169.16.141.17.3.96
130 DATA 0.0 0.0.0.32 247.183.160.1.177 20
140 DATA 72.136.177.20 168.104.32.145.179.32
150 DATA 43 188 240.9.144.7 169 48.160.64 32
160 DATA 103,184,96,0,145,0,0,0,0
```

La seconda risolve invece un problema più complesso e cioè la valutazione di un'espressione algebrica contenuta in una stringa: può essere utile quando s'intendono tracciare grafici di funzioni e si desidera immettere la funzione stessa (espressa per esempio come $y = f(x)$) da tastiera. Come nel caso precedente, il programma Basic installa la routine in memoria e poi attiva la funzione USR() tramite la SYS 24576. A questo punto, se la variabile stringa A$ contiene l'espressione algebrica "X]2 – 3" e la variabile X contiene il valore 5, eseguendo l'istruzione `A = USR(A$)` la variabile A assume il risultato dell'operazione, cioè 22. Ovviamente, l'argomento della funzione USR() può essere una qualsiasi variabile stringa o una stringa (per esempio, `USR("X]2-3")`) contenente un'espressione algebrica che il Basic può calcolare.

```
100 FOR I=24576 TO 24659:READA:C=C+A:POKEI,A:NEXT
```

Input/output rivela ogni mese ai suoi lettori routine di programmazione per C-64, C-128 e Amiga. Avete in mente un'idea particolare, una routine, un utile stratagemma di programmazione, avete sviluppato un breve programma o in generale pensate che le Vostre conoscenze possano interessare l'utenza Commodore? Allora scrivete, e inviate gli eventuali listati stampati su carta bianca o salvati su disco a:

COMMODORE GAZETTE
Input/output
Via Monte Napoleone,9
20121 - Milano

```
110 IF C<>9744 THENPRINT"ERRORE NEI DATA...":END
115 SYS24576
120 DATA 169,96,141,18,3,169,16,141,17,3,96
130 DATA 0,0,0,0,0,134,251,132,252,165,122
140 DATA 133,253,165,123,133,254,32,130,183
150 DATA 208,3,76,247,184,192,80,48,3,76,72
160 DATA 178,169,0,153,0,2,136,177,34,153,0
170 DATA 2,136,16,248,169,0,133,122,169,2,133
180 DATA 123,32,124,165,32,166,173,165,253
190 DATA 133,122,165,254,133,123,166,251,164
200 DATA 252,96
```

Si noti che le due routine si trovano in memoria a due indirizzi diversi, al fine di permetterne la coesistenza.

Commodore 128

***198 Un sostituto per la chiocciolina -** Nel sistema operativo del C-128, così come in quello del C-64, quando si cerca di sovrascrivere un file già esistente non si ottiene la cancellazione automatica del contenuto precedente, ma viene generato un messaggio d'errore. Esiste però la possibilità di ottenere questa funzione, pericolosa ma talvolta utile: basta anteporre il carattere "@" al nome del file che si vuole salvare. Il problema è che a causa di un bug nella ROM del 1541 si rischia di perdere il file e anche l'intero disco. Abbiamo quindi pensato di collegare alla routine di SAVE del kernel del C-128 una routine che controlla l'eventuale esistenza del file, e se è presente avvisa l'utente con un messaggio. L'utente può quindi confermare la richiesta di scrittura, nel qual caso il file viene cancellato con il comando "S0" del disk drive, oppure può annullare l'operazione. Si noti che tutto questo accade in modo diretto, mentre in modo programma il file viene sempre cancellato senza alcuna richiesta di conferma, conformemente agli standard del Basic 7.0.

```
10 REM ****************************
20 REM **     SCRATCH & SAVE     **
30 REM ****************************
40 :
50 FORI=3584TO3812:READA:C=C+A:POKEI,A:NEXT
60 IFC<>29660THENPRINT"ERRORE NEI DATA...":END
70 POKEDEC("332"),11:POKEDEC("333"),14
80 :
100 DATA 32,204,255,169,15,32,195,255,76,78
105 DATA 245,169,1,162,8,168,32,186,255,32
110 DATA 192,255,169,15,168,162,8,32,186,255
120 DATA 165,183,72,169,0,133,183,32,192,255
130 DATA 104,133,183,162,15,32,198,255,32,207
140 DATA 255,201,54,208,201,32,207,255,201
150 DATA 51,208,194,169,1,32,195,255,32,204
160 DATA 255,169,15,32,195,255,165,157,240
170 DATA 53,32,125,255,73,76,32,70,73,76,69
180 DATA 32,69,83,73,83,84,69,44,32,80,82,79
190 DATA 67,69,68,79,63,32,40,83,47,78,41,0
```


SoftMail

VENDITA PER CORRISPONDENZA DI PROGRAMMI ORIGINALI PER TUTTI I TIPI DI COMPUTERS

VIA NAPOLEONA 16 - 22100 COMO - TEL. (031) 30.01.74

® SoftMail è un marchio registrato da Lago snc

SOFT MAIL

Vuoi ricevere il nostro **catalogo gratuito a colori?**
Richiedilo allo (031) 300.174

Attenzione!

Sei già cliente SoftMail?

Chiusi per ferie dal 1 al 15 agosto.

Allora rimani in stretto contatto: da settembre scattano GRANDIOSE iniziative!

ACCESSORI

Copritastiera A500	25.000
Final cartdrige III	110.000
Flicker master	29.000
Joy. SpeedKing A.F.	33.000
Joy. Tac 5	39.000
MouseMat tappetino	22.500
Coprimouse	20.000
Portamouse	12.500
Portadischi 3" (30)	34.000
Portadischi 5" (40)	37.000
SlimLine (tastiera64)	49.000
Tappetino mouse	22.500

LIBRI/HINTS & TIPS

Bard's tale : vari	tel.
Deathlord	19.000
Dungeon master	25.000
Elite	18.000
Mars saga	19.000
Pool of radiance	20.000
Quest for clues	39.000
Sierra on line: vari	tel.
Ultima: vari	tel.
Wasteland	16.500

AMIGA

Aegis draw 2000	275.000
Airborne ranger	59.000
ANIMagic	tel.
Audiomaster II	105.000
Bal.of power 1990	49.000
Barbarian II	39.000
Battlehawks	29.000
Bio challenge	39.000
Castle warrior	tel.
Cosmic pirate ITA	39.000
Deja vu II	49.000
Deluxe photolab	250.000
Deluxe paint III	tel.
Digiview gold (PAL)	330.000
Disk drive esterno	299.000
Dragon's lair (1 Mb)	75.000
Driller ITALIANO	59.000
Dungeonmaster1MB	59.000
Elite	45.000
F16 Falcon	59.000
Fire brigade (1 Mb)	85.000
Flight simulator II	99.000
Scenery disks nuovi	tel.
Graphics Starter Kit	tel.
Hound of shadow	tel.
Kick off	29.000
LightCamtActionI	99.000
Lord ... rising sun	69.000
Microprose soccer	49.000
Millennium 2.2	49.000
Modeler 3D	129.000
Neuromancer	tel.
Photon paint 2.0	215.000
Photon video:	
Cell animator	199.000
Transp.controller	490.000
Populous	55.000
Data disks	tel.
Powerdrome	tel.
Read heat Danko	29.000
Renegade	39.000
Rocket ranger	59.000
S.E.U.C.K. ITA	49.000
Space harrier	29.000
Sword of sodan	69.000
Test Drive II	tel.
Scenery disks	tel.
TV sport football	59.000
Virus killer ITA	29.000
Winter edition	25.000
Workbench 1.3	45.000
Zak McKracken ITA	tel.
Zany golf	59.000
Zoetrope	199.000

Tony Severa's hint disks

Bard's tale	*29.000*
Heroes of the lance	*29.000*
Leisure suit Larry	*19.000*

CBM 64/128 DISCO

Accolade tutti gli sports	*tel.*
Adv.dungeons&dragons tel.	
Barbarian II	25.000
Battles of Napoleon	59.000
Defender ... Crown	15.000
Ferrari F1	tel.
Fire King	tel.
F1 Manager	tel.
F14 Tomcat	29.000
Grand prix circuit	29.000
Italy 90 soccer	25.000
Legend blacksilver	18.000
Mars saga	35.000
McArthur's war	49.000
Microprose soccer	49.000
Neuromancer	39.000
Project firestart	39.000
Red heat Danko	18.000
Renegade III	18.000
Rocket ranger	39.000
R-Type	25.000
S.E.U.C.K. ITA	35.000
Test Drive II	tel.
Ultima V	49.000
Wasteland	39.000
Zak McKracken ITA	tel.

ORA APERTI AL PUBBLICO

Invia **subito** il buono d'ordine e riceverai insieme alla merce uno **speciale aggiornamento** con **tantissime offerte a prezzi incredibili!** *Non perdere la stagione dei veri affari!*

Buono d'ordine da inviare a: LAGO DIVISIONE SOFTMAIL, VIA NAPOLEONA 16, 22100 COMO, TEL. (031) 30.01.74, FAX (031) 30.02.14

Desidero ricevere i seguenti articoli:

Titolo del programma	Computer	Prezzo
G	Spese di spedizione Lit.	5.000
ORDINE MINIMO LIT.25.000 (SPESE ESCLUSE)	TOTALE LIT.	

☐ Pagherò al postino in contrassegno

Addebitate l'importo sulla mia: ☐ CartaSI ☐ Mastercard ☐ Visa ☐ American Express

Numero _____ scad. _____

Cognome e nome _____

Indirizzo _____ Nr. _____

CAP _____ Città _____ Prov. ____ Tel. _____

FIRMA (Se minorenne quella di un genitore)

Verranno evasi SOLO gli ordini firmati _____

```
200 DATA 165,213,201,13,240,15,201,39,240,5
210 DATA 32,225,255,208,241,169,0,133,208,24
220 DATA 96,169,0,133,208,230,183,165,188,201
230 DATA 17,208,13,169,83,141,255,16,169,16
240 DATA 133,188,169,255,208,21,230,183,230
250 DATA 183,169,83,141,125,10,169,48,141,126
260 DATA 10,169,58,141,127,10,169,125,133,187
270 DATA 169,15,133,184,133,185,32,192,255
280 DATA 169,15,32,195,255,198,183,169,0,133
290 DATA 184,133,185,165,188,201,10,240,8,169
300 DATA 17,133,188,169,0,240,6,198,183,198
310 DATA 183,169,128,133,187,76,78,245
```

Il funzionamento della routine può essere così riassunto: utilizzando il nome del file già presente in memoria (passato dal Basic al kernel), si tenta di aprirlo in scrittura. Se il disk drive restituisce un errore 63 (file exists) allora si prendono i provvedimenti descritti poco sopra, altrimenti si prosegue con il normale salvataggio del file. Il programma viene allocato a partire dalla locazione $E00, ossia nell'area di definizione degli sprite: è comunque completamente rilocabile senza interventi, eccetto la modifica dei due comandi POKE alla linea 70, che dovranno indicare l'indirizzo (incrementato di 11) a partire dal quale dev'essere collocata in memoria la routine.

Commodore Amiga

***199 Un lampeggiatore in background -** Il led indicatore dell'accensione dell'Amiga ha una notevole proprietà: a differenza di quanto accadeva nei precedenti computer Commodore, è controllabile via software. Abbiamo pensato di sfruttare questa caratteristica per fornire un'indicazione sullo stato di un task in background. In pratica ci baseremo sull'indicazione fornita da questo led per sapere se un task mandato in esecuzione in precedenza sta ancora compiendo il suo lavoro o è terminato.

Il corpo principale del programma inserisce nel sistema un secondo task, il task in background, il quale provvede a far lampeggiare per un certo periodo di tempo il led rosso dell'Amiga, così da segnalare la sua attività. Nel frattempo, il corpo principale del programma attende che l'utente prema un tasto, nel qual caso provvede a chiudere il task in background qualora risulti ancora aperto. Il sorgente in C è il seguente.

```
#include <exec/types.h>
#include <exec/tasks.h>
#include <exec/exec.h>

struct Task *CreateTask();
struct Task *TaskPtr;

char        ret;
void MioTask()
{
   ULONG count;
   for (count = 0L; count < 0xfffff; count++)
   {
      if (!(count % 5000L))
      {
#asm
      bchg.b #1,$bfe001
#endasm
      }
   }
#asm
   bclr    #1,$bfe001
#endasm
}

main()
{
 TaskPtr = CreateTask("MioTask",0L,MioTask,100L);
 printf("&MioTask 0x%lx\n",&MioTask);
 printf("SPLower  0x%lx\n",TaskPtr->tc_SPLower);
 printf("SPUpper  0x%lx\n",TaskPtr->tc_SPUpper);
 printf("SPReg    0x%lx\n",TaskPtr->tc_SPReg);
 printf("Type     %d\n",TaskPtr->tc_Node.ln_Type);
 printf("Pri      %d\n",TaskPtr->tc_Node.ln_Pri);
 printf("Name     %s\n",TaskPtr->tc_Node.ln_Name);
 puts("Premi un tasto per chiudere il task");
 read(ret);

 Disable();
 if (FindTask("MioTask"))
 {
    DeleteTask(TaskPtr);
#asm
    bclr    #1,$bfe001
#endasm
 }
 Enable();
 exit(FALSE);
}
```

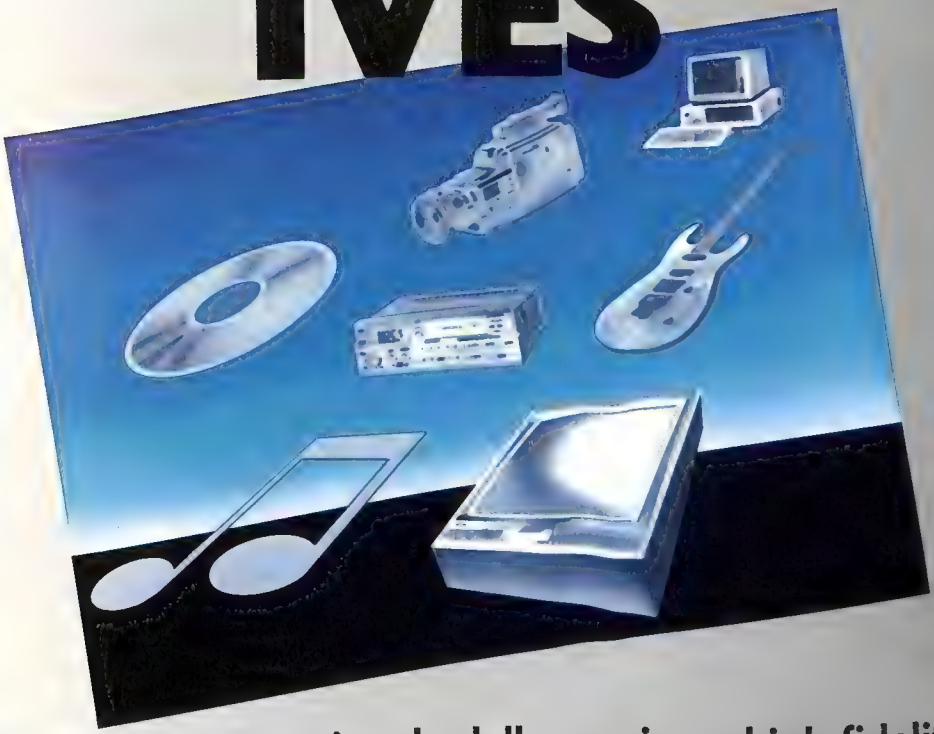
Il controllo del led avviene tramite il bit meno significativo della locazione di memoria 0xBFE001: quando questo bit è a livello basso, il led è acceso, e viceversa.

Questo sistema per la verifica del funzionamento dei task in background è utile sia che si voglia imparare a inserire un task nel sistema, sia che si voglia effettuare il debug di particolari task privi di output a schermo.

Infine, si noterà che la frequenza con cui il led lampeggia è pesantemente influenzata dal cosiddetto overhead di sistema, cioè dal numero di task attivi nel sistema. Se durante il lampeggio, per esempio, si muove una finestra, si nota una diminuizione della frequenza dovuta al fatto che il nostro task riceve meno partizioni di tempo della CPU. In quest'ottica, il task che il programma alloca può costituire un simpatico strumento di rilevamento dell'overhead di sistema.

■

SIM-HI·FI IVES



23º salone internazionale della musica e high fidelity international video and consumer electronics show

14-18 settembre 1989
Fiera Milano

STRUMENTI MUSICALI,
ALTA FEDELTÀ, HOME VIDEO,
HI-FI CAR, CAR ALARM SYSTEMS,
PERSONAL COMPUTER, TV,
VIDEOREGISTRAZIONE,
ELETTRONICA DI CONSUMO.

ASSOEXPO

SIM HI·FI

un grande Sim!

HOME VIDEO

VIVA i giovani

Ingressi: Piazza Carlo Magno - Via Gattamelata - **Orario:** 9.00-18.00
Aperto al pubblico: 14•15•16•17 - **Giornata professionale:** lunedì 18 settembre

ASSOCIATO ASSOMOSTRE

Segreteria Generale SIM-HI•FI-IVES: Via Domenichino, 11 - 20149 Milano - Tel.: 02-4815541 - Telex: 313627 - Fax 02-4980330

FIERA MILANO

L'AMIGA NELLE UNIVERSITÀ AMERICANE

Il nostro giro d'orizzonte negli istituti di ricerca statunitensi mette in luce alcuni originali impieghi dell'Amiga, il potenziale computer scientifico del futuro

di Matthew Leeds

Per molti l'Amiga è il più eccezionale computer da gioco che sia mai stato inventato, mentre per altri rappresenta il logico erede del C-64 come perfetto home computer. Un'opinione forse meno conosciuta, ma più importante per il successo a lungo termine dell'Amiga, è quella dei ricercatori che l'hanno introdotto nella comunità scientifica e universitaria statunitense. L'Amiga viene impiegato per risolvere i misteri della medicina, per esplorare gli intimi segreti dell'atomo, per migliorare la nostra comprensione dei meccanismi del corpo umano e per esplorare la Terra su cui viviamo. In ciascun caso il commento di chi ne fa uso è sempre lo stesso: l'Amiga mette a disposizione potenza di calcolo e funzionalità a un prezzo molto inferiore a quello richiesto dai metodi tradizionali. In questo articolo presentiamo sei testimonianze dei casi in cui l'Amiga dà il suo contributo nella soluzione di problemi scientifici di vasta portata.

Fisica nucleare

Marvin Wienstein, fisico (Laboratorio dell'acceleratore lineare di Stanford).

«Questo è un laboratorio statale e una parte del nostro lavoro consiste nella preparazione di relazioni riguardanti le ricerche che vi si svolgono: l'Amiga ci serve soprattutto per il word processing. Fondamentalmente, quello di cui avevamo bisogno era una macchina che ci consentisse la preview di testi redatti con un editor e trasformati in pagine grafiche con *TeX* (un elaborato programma d'impaginazione): non c'è niente che sia in grado di gestire simboli matematici ed equazioni come fa *TeX*. Il problema è che il mainframe su cui viene mandato in esecuzione non è in grado di fornirci una preview dei testi nel loro formato grafico finale. Avevamo quindi bisogno di qualcosa che ci consentisse di utilizzare *TeX* sul mainframe, che ci mettesse a disposizione un buon terminale a colori e che avesse anche capacità di elaborazione proprie. Attualmente ci serviamo soltanto di A2000, quasi tutti dotati di hard disk, con schede FlickerFixer (per l'eliminazione del flickering, n.d.r.) e monitor Zenith.

Un'altra cosa importante è che *REXX*, cioè il linguaggio che usiamo sui mainframe, è disponibile anche per l'Amiga, e quindi possiamo trasferire su questo computer la maggior parte del materiale elaborato con i grandi sistemi. Abbiamo per esempio programmi di disegno interattivi che funzionano sul mainframe e che aprono una finestra sull'Amiga per dotare il sistema di un'interfaccia grafica. A questo scopo, l'Amiga emula un terminale che può ricevere comandi *REXX* dai programmi attivi sul mainframe. Quando il programma "terminale" riceve un comando *REXX*, lo passa direttamente al *REXX* installato sull'Amiga. Tutto funziona correttamente grazie alle capacità multitasking dell'Amiga che consentono a diversi programmi contemporaneamente in funzione di avere message port tramite le quali ricevere dati e comandi da altri programmi.

Al mainframe accedono da 1.500 a 2.000 utenti. Non c'è modo di realizzare direttamente un'interfaccia grafica per le applicazioni: se ci provassimo l'intero sistema si bloccherebbe. In pratica ci limitiamo a fare in modo che il mainframe dica all'Amiga di aprire finestre, prendersi cura delle pressioni sul pulsante del mouse, gestire icone e infine inviare al mainframe un segnale di ritorno quando ha finito il

lavoro. In altre parole, l'Amiga si comporta da terminale "intelligente" che offre un'interfaccia grafica al mainframe. Non si nota per nulla che si sta utilizzando l'Amiga come un terminale, anzi, in un certo senso si potrebbe dire che il mainframe funziona come un coprocessore per l'Amiga.

Gli Amiga sono connessi tramite la porta seriale e permettono di realizzare una rete con una spesa molto bassa. Non c'è bisogno di tutta quella tecnologia "intelligente" (protocolli per i file, invio di pacchetti, capacità decisionali...) di cui ha bisogno una rete tradizionale come Ethernet. Grazie alle capacità multitasking possiamo semplicemente comunicare all'Amiga che deve inviare o ricevere dati, ed è il computer stesso che si occupa di attendere il momento più opportuno. Facciamo un esempio in cui ci sono tre applicazioni separate attive contemporaneamente: l'editor di *TeX* e lo stesso *TeX* sul mainframe e il programma di preview sull'Amiga, tutti e tre collegati tramite *REXX*. Per quanto riguarda l'utente, è come se l'intero complesso fosse un solo programma in funzione sull'Amiga.

L'utente entra nell'editor, crea un documento, torna al menu e passa a *TeX* questo documento perché venga trasformato in un testo grafico definitivo. Il sistema verifica se *TeX* sul mainframe e il preview sull'Amiga sono attivi e se non lo sono li carica, quindi fa all'utente alcune domande tramite un requester, e alla fine ordina a *TeX* di elaborare il testo e inviare il risultato al programma di preview, il quale sfruttando la grafica dell'Amiga visualizza il documento nel suo aspetto finale. Il tutto come se si trattasse di un'unica applicazione molto sofisticata. Se sul mercato dovesse comparire un editor migliore potremmo tranquillamente sostituire quello che usiamo adesso: la nostra è una tecnica "modulare" che ci permette di costruire l'ambiente di lavoro nel modo che si avvicina maggiormente alle nostre esigenze particolari.

Sullo sfondo il famoso campanile dell'Università della California, a Berkeley

Al laboratorio abbiamo circa 50 Amiga, ma presto ne arriveranno altri cento. L'Amiga è meno costoso di un terminale grafico a colori, e d'altra parte terminali grafici a colori che contemporaneamente siano anche multitasking non ne esistono a nessun prezzo».

Radiologia

William Bennett, professore aggiunto e dottore in medicina, e *Robert McGhee jr., dottore in medicina* (Università dello Stato dell'Ohio, dipartimento di radiologia).

«Ci serviamo dell'Amiga per una serie di programmi didattici sull'argomento della radiologia. L'idea è di presentare diverse immagini digitalizzate, ognuna accompagnata da una domanda e da alcune possibili risposte. Quando lo studente seleziona la risposta esatta gli vengono fornite ulteriori informazioni sull'argomento e una nuova immagine con le corrette caratteristiche evidenziate tramite "etichette", mentre in caso di errore il programma comunica che la risposta è sbagliata e spiega perché, quindi offre un'altra possibilità di scelta.

Abbiamo creato le immagini per questi programmi didattici ("tutorial") utilizzando *Digiview* della NewTek per le digitalizzazioni, e aggiungendo poi le etichette con un programma di disegno. I tutorial sono stati invece realizzati usando *The Director*. L'Amiga offre la possibilità di un insegnamento interattivo senza bisogno di realizzare complicati programmi o di ricorrere a filmini, videodischi o altro materiale. In pratica offre un completo ambiente interattivo e parallelamente dà gli strumenti per la creazione di elaborati tutorial».

Sismologia

Carl Johnson e *Tom English* (Osservatorio Vulcanico del Parco nazionale Volcano delle Hawaii).

«L'Amiga ci viene in aiuto per dare una forma coerente all'impressionante quantità di dati che raccogliamo sull'attività sismica. Vi sono oltre 70 punti di raccolta di dati sismografici nell'isola. Tutti i dati grezzi vengono riversati in due computer VAX-75, che li trasformano in una descrizione vettoriale dei movimenti della crosta terrestre. I vettori sono quindi inviati all'Amiga che li sovrappone a una mappa delle Hawaii.

Avevamo bisogno di qualcosa che potesse sostenere il flusso di dati proveniente dai VAX, convertendolo in una rappresentazione grafica in tempo reale. Volevamo inoltre poter rappresentare i dati in diverse forme, e l'Amiga si è dimostrato perfetto per entrambe le esigenze. È abbastanza veloce da tenere il passo dei VAX, ed essendo multitasking ci permette di accedere contemporaneamente a diversi insiemi di dati.

Usiamo l'Amiga anche per realizzare simulazioni di eruzioni vulcaniche e prevedere la direzione dei flussi di lava sulla base dei dati noti. Con un po' di fortuna potremmo riuscire a costruire accurati modelli dei flussi lavici e avvertire per tempo la popolazione durante le eruzioni. L'Amiga è un computer stupefacente, ed è stato una scelta ovvia per questo tipo di applicazioni».

Ecologia

Andrew Glick della Justus Engineering (consulente del Centro naturalistico McCurdy).

Il Centro McCurdy ha effettuato uno studio sui serpenti a sonagli per capire in quale misura i cambiamenti nelle loro abitudini alimentari potevano dare indicazioni sulle alterazioni ecologiche dovute all'intervento umano. Il processo digestivo nello stomaco dei serpenti rende particolarmente difficile l'identificazione del contenuto. Glick si è servito di un Amiga 2000 con un digitalizzatore video e relativo software, per analizzare alcune fotografie del contenuto dello stomaco del serpenti.

«Il software di gestione delle immagini è un programma commerciale, *PIXmate*. Lo utilizziamo per alterare le immagini allo scopo di trarne tutte le possibili informazioni. Si comincia digitalizzando le fotografie e trasferendo le immagini in *PIXmate*, poi si esalta il contrasto tramite un processo chiamato "incremento dei contrasti minimi" per evidenziare i particolari del contenuto dello stomaco; le parti ossee dell'animale sono invece segnalate tramite falsi colori e l'intero processo richiede meno di sei ore.

L'Amiga è decisamente meno costoso dei pacchetti grafici disponibili per i PC, e la possibilità di trasferire dati IFF da un'applicazione all'altra rende particolarmente semplice il nostro tipo di lavoro. Con gli altri sistemi, è necessario modificare i dati per usarli con applicazioni diverse e a volte nel processo si perdono dettagli importanti».

Fisiologia e controllo di qualità

John Davis (Optimum Comparative Technologies).

«Era da parecchio tempo che cercavo un sistema per quantificare i cambiamenti subiti dal corpo umano in seguito a terapie fisiche. La via che seguo ora consiste nel prendere fotografie prima e dopo la cura, una tecnica che viene chiamata OptiComp. L'Amiga confronta le due immagini e ne mostra le differenze con quella che potremmo chiamare "rappresentazione dinamica alternata". Poiché opera a 32 bit, si possono fare alcune trasformazioni sull'immagine, come per esempio la rimozione del tessuto epiteliale e la rappresentazione della struttura ossea o muscolare. Si può esaminare come i muscoli sono attaccati alle ossa, e analizzare particolari che non si vedono neppure nelle radiografie. Con una radiografia, o anche con la tecnica della risonanza magnetica nucleare, non si può capire facilmente che cos'è accaduto ai tessuti molli, mentre con questo sistema si riescono a ottenere risultati interessanti.

L'Università della Virginia e il dipartimento Motion Sciences del Lynchberg College hanno ordinato alcuni esemplari del sistema e la squadra olimpica di pattinaggio intende utilizzarlo per gli allenamenti. Prima non c'era nessun metodo oggettivo per misurare i risultati ottenuti con discipline come la chiropratica: quando un osservatore rilevava un allungamento nella regione lombare o un allargamento della regione toracica, ce n'era sempre un altro che non vedeva nessuna variazione. Il nostro sistema elimina ogni incertezza provvedendo misure quantitative dei risultati ottenuti con il trattamento.

Si possono persino individuare i danni subiti dai tessuti molli. Per esempio si può individuare una costola staccata dalla cartilagine nello sterno. In realtà *è possibile* fare una radiografia del tessuto molle, ma è estremamente complicato, perché è necessario iniettare sostanze colorate che producano l'opportuno contrasto e alcuni sono allergici a queste sostanze. Con il mio sistema invece si scandisce elettronicamente il soggetto, si realizza un'immagine digitale e la si elabora con l'Amiga fino a ottenere una "mappa" dettagliatissima della zona interessata. La mappa mostra esattamente dove potrebbe essere il punto di separazione. Si arriva a vedere la ragnatela di tessuto che circonda l'area interessata e perfino la stessa rottura.

Queste sono state le nostre prime utilizzazioni dell'Amiga. In seguito, dopo aver lavorato per più di due anni e mezzo sul corpo umano (una struttura in continua evoluzione e particolarmente difficile da analizzare con immagini statiche), abbiamo ritenuto che lo

stesso sistema sarebbe stato adattissimo per esaminare oggetti statici. Attualmente stiamo lavorando con la Continental Airlines per realizzare una "imaging station" nella zona dell'aeroporto in cui gli aeroplani rullano prima del decollo. Il computer esaminerebbe il rivestimento esterno dell'aereo utilizzando una tecnica di analisi sottrattiva digitale, e sarebbe in grado d'individuare ogni minima differenza sopravvenuta dopo l'esame precedente. Per esempio l'aeroplano potrebbe aver avuto un atterraggio difficile, e potrebbe esserci stato qualche danno ai punti di collegamento tra rivestimento esterno e struttura portante, e alcuni danni potrebbero non essere individuabili mediante la normale ispezione visiva. O semplicemente il portello di un aereo da trasporto potrebbe non essere perfettamente chiuso. Il sistema sarà in grado d'individuare tutti i problemi di questo tipo e avvertire una squadra d'ispezione.

Il ministero della Difesa, la FAA (Federal Aviation Administration) dello Stato di Washington e la Continental Airlines sono molto interessati a questa idea. Ogni postazione non costa più di 25 mila dollari, dal momento che non è necessario né un computer mainframe né la presenza costante di un operatore. Il sistema mantiene in memoria la "storia" di ogni aeroplano e può individuare un bullone allentato o mancante, tracce di corrosione e persino la presenza di sporco o ghiaccio sulla superficie del velivolo. Un'ispezione umana impiegherebbe anni a scoprire alcuni dei difetti che questa tecnica ci permette di rilevare immediatamente.

Il sistema di controllo che abbiamo elaborato comprende un monitor a colori ad alta risoluzione da 19 pollici, una telecamera ad alta risoluzione, un digitalizzatore in tempo reale con una scala di grigi da 256 tonalità, e infine un Amiga 2000 dotato di un hard disk da 100 MB o di un disco ottico. Sia il software per l'elaborazione delle immagini sia quello per mantenere traccia delle immagini elaborate dal sistema è stato scritto appositamente per le nostre esigenze.

L'Amiga ha capacità incredibili per quanto riguarda l'elaborazione delle immagini. Ora stiamo prendendo in considerazione anche l'idea di un sistema d'allarme che controlli se in una certa zona avviene un qualsiasi movimento: se individua qualcosa, scatta una fotografia, disinserisce il sistema d'allarme e avverte chi di dovere».

Anatomia

Ronald M. Harper, professore di anatomia (UCLA).

«Noi utilizziamo l'Amiga per dare una rappresentazione grafica ai dati ottenuti con la risonanza magnetica nucleare, con il TAC, con l'angiografia digitale a sottrazione e con la tomografia a elettroni-positroni (PET). Ma il campo in cui utilizziamo in più larga misura l'Amiga è quello dell'acquisizione e dell'analisi di dati provenienti da registrazioni di lungo periodo sull'attività cardiaca, respiratoria e cerebrale durante il sonno. I dati di questo tipo riguardano soprattutto neonati a rischio, e pazienti epilettici o cardiopatici.

Un convertitore analogico-digitale a più canali è collegato all'Amiga per raccogliere i dati, i quali vengono esaminati da alcune routine di analisi che ne forniscono una rappresentazione visiva. Prima di utilizzare l'Amiga ricorrevamo al PDP-11, con il quale ottenevamo risultati analoghi ma a un costo dieci volte superiore. L'Amiga è anche dieci volte meno ingombrante di un PDP-11 e dieci volte più facile da usare per le infermiere e il personale medico grazie alla semplicità della sua interfaccia grafica.

Spesso ricorriamo all'Amiga anche per la creazione di presentazioni da utilizzare durante i convegni. In genere per la visualizzazione delle informazioni digitali utilizziamo un programma come *Snip* della Digital Dynamics, poi trasformiamo le immagini in file grafici e le trasferiamo in *Deluxe Paint* o in *Express Paint* per aggiungere testi e didascalie, quindi trasformiamo le immagini in diapositive da 35 mm grazie alla Polaroid Palette.

Con l'Amiga riusciamo ad analizzare dati di tipo estremamente diverso, come intervalli cardiaci ed elettroencefalogrammi, e per presentarli ci serviamo di grafici, istogrammi, diagrammi scatter (a nube di punti) e così via.

L'Amiga è la macchina ideale per la presentazione grafica dei dati e fornisce prestazioni straordinarie a un costo praticamente nullo: una scheda grafica RGB 512 x 512 destinata al PDP-11 costa quasi quanto un Amiga. E nonostante la differenza di prezzo, anche i programmi disponibili per questa macchina sono di ottimo livello. Per esempio ci serviamo di *J-Forth*, un linguaggio che permette di sviluppare programmi molto rapidamente ed è ideale per l'analisi dei dati in tempo reale.

Il multitasking, poi, è una caratteristica inestimabile in un laboratorio. C'è un processore che raccoglie i dati e l'Amiga non deve fare altro che riceverli e trasferirli sulla memoria di massa: questo lascia un sacco di tempo libero alla CPU. Uno dei modi in cui sfruttiamo questo tempo è l'invio di un codice di temporizzazione a un dispositivo poligrafico (un registratore analogico) mentre i dati si riversano nell'hard disk, permettendoci di sincronizzarli. In passato serviva un dispositivo molto costoso solo per il codice di temporizzazione.

Siamo passati all'Amiga due anni fa. Se avessimo acquistato alcune decine di computer di qualunque altro tipo avremmo ben presto superato il nostro budget. Con l'Amiga posso perfino permettermi di tenerne uno a casa e, se voglio, continuare a lavorare di notte o durante il fine settimana senza essere costretto a fermarmi al laboratorio». ■

L'ULTIMO NATO TRA I PC COMMODORE: IL PC40-III

Un compatibile IBM AT "turbo", con una buona potenza di calcolo. Di serie la grafica VGA, un hard disk da 40 MB e un monitor bi-sync a fosfori bianchi

di Dario Greggio

A distanza di un anno dalla prova del penultimo PC della terza serie, constatiamo che per i computer MS-DOS della Commodore continua un processo di rinnovamento radicale. I computer della prima serie entravano in un mercato che per alcuni anni aveva visto solo PC IBM "base" e qualche clone del tutto uguale agli originali. Le prestazioni delle varie macchine erano più o meno dello stesso livello e i PC Commodore, nel bene e nel male, non si distinguevano dagli altri. Ma i prezzi sempre più allettanti causarono una diffusione a macchia d'olio dei "compatibili" che colse impreparati quasi tutti gli operatori. Non tardò molto perché il successo innescasse un vero e proprio boom, una corsa febbrile alla prestazione da capogiro.

Oggi siamo nel pieno di questa guerra tecnologica. Assistiamo a una vera e propria escalation di record (clock, memoria RAM, periferiche) che sopravvivono a malapena il tempo di apparire sulla stampa specializzata. Il confronto fra le nuove CPU, l'incremento continuo della quantità di memoria offerta (che tra l'altro per lungo tempo ha avuto prezzi bassissimi, anche se ora non è più così), e il proliferare di schede aggiuntive di ogni genere sono le caratteristiche di questa gara tra i vari PC. C'è indubbiamente molta attività nel settore e alcuni osservatori la giudicano senza mezzi termini una corsa insensata, soprattutto pensando che l'MS-DOS è nato nel 1980, e che CPU avanzate come l'Intel 80386 sono davvero sprecate in un ambiente che oggi è senza dubbio obsoleto. In un panorama come questo, comunque, i produttori di PC devono cercare di tenersi al passo, conquistando possibilmente le posizioni migliori.

La prima serie PC della Commodore non è ovviamente rimasta estranea a questo processo evolutivo, trasformandosi nella seconda e poi nell'attuale terza serie. In questo articolo analizzeremo l'AT PC40-III: prestazioni naturalmente superiori, ma anche ergonomicità e design più curati, a dimostrare il crescente interesse verso questa fascia di mercato da parte della Commodore.

Se il PC10-III si distingueva per le sue tre frequenze di clock, per la tastiera avanzata modello AT e per la sofisticata scheda grafica, il PC40-III è un AT compatibile dotato del noto microprocessore Intel 80286 funzionante a 12 MHz, 1 MB di RAM e un hard disk da 40 MB e 28 ms di tempo d'accesso medio. E tutto questo a un prezzo di poco superiore rispetto a quello del precedente PC40-40, ormai uscito di produzione. La potenza di queste macchine è davvero notevole e l'impiego delle applicazioni in ambiente MS-DOS non ne utilizza che una piccola parte.

Dall'esterno

Il PC40-III si presenta con un cabinet di dimensioni decisamente ridotte rispetto ai suoi pachidermici predecessori. L'area di base è infatti la stessa del PC10 ultimo modello e come si può facilmente indovinare, il miracolo delle dimensioni così ridotte è stato possibile integrando buona parte della logica di controllo dei bus in alcuni grossi chip a 84 pin. Rimandandovi ai prossimi paragrafi per i particolari tecnici, dobbiamo dire subito che aver dimezzato l'ingombro del computer dovrebbe essere un fatto positivo per tutti, anche per chi ha a disposizione tutto lo spazio che desidera...

Due colori si alternano sul cabinet, sulla tastiera e sul moni-

tor, e si tratta dei consueti toni di grigio-beige della Commodore. Spicca invece il marchio Commodore, che a partire dalla serie III ha ripreso gli antichi colori dei tempi del C-64 ed è stato incastonato in una gemma luccicante. Due led collocati sotto il marchio stanno a indicare la presenza della tensione di alimentazione e l'attività dell'hard disk. Sulla destra troviamo poi un disk drive da 5,25" ad alta capacità (1,2 MB) e, sotto, lo spazio per aggiungere un'altra memoria di massa (un altro disk drive o magari uno streamer).

L'aspetto generale è molto gradevole, sia per l'accostamento delle due tonalità di colore, sia per il fatto che il pannello anteriore è inclinato verso l'utente; ad aumentare poi questa sensazione viene la tastiera. Si tratta del modello ormai standard, originariamente introdotto dalla IBM per i computer PS/2: dispone di 102 tasti sensibili ma precisi grazie allo scatto che si avverte a metà pressione. I tasti sono situati in una struttura leggermente concava, allo scopo di facilitare la digitazione; per lo stesso motivo, piccole sporgenze evidenziano i tasti F, J e 5 (quest'ultimo sulla tastierina numerica).

Sul lato frontale del cabinet troviamo ancora la serratura per la protezione della propria postazione di lavoro. Passiamo poi al lato destro, dove viene collegata la tastiera con un consueto connettore DIN a cinque poli e dove troviamo anche un insolito tastino di reset. Sul retro si trovano invece diverse porte: oltre alle feritoie corrispondenti agli slot per le schede di espansione, abbiamo il connettore femmina per la porta parallela e quello maschio per la seriale (entrambi di tipo DB-25); c'è poi una finestrella che rende possibile l'accesso ai dip switch di configurazione della scheda grafica, cosa di cui parleremo più avanti. Notiamo solo che, molto opportunamente, una striscia adesiva riporta le possibili configurazioni relative ai dip switch, per evitare di dover ricorrere continuamente al manuale.

Segue quindi il connettore per il monitor VGA, un tipo ancora poco comune ma che dovrebbe diventare uno standard quando le schede grafiche VGA (e i relativi monitor) si saranno diffuse a sufficienza. Si tratta di un Cannon DB-9 con tre file di piedini anziché due. Ovviamente, l'impiego di questo connettore rende più difficile sostituire il monitor in dotazione alla macchina con altri monitor multisync di marche diverse. Per ultimi, a sinistra del Cannon, troviamo una boccola RCA per l'audio e un connettore standard per il mouse.

Il monitor fornito con il computer è il 1403, 14" bi-sync a fosfori bianco-carta. Approviamo l'idea di fornire questo tipo di monitor, visto che l'adattatore grafico contenuto nel computer mette a disposizione una risoluzione video idonea alle applicazioni più spiccatamente grafiche (come il desktop publishing), per le quali un monitor a fosfori bianchi è più che indicato. Purtroppo però la qualità del tubo catodico non sembra particolarmente elevata in quanto nei modi grafici con maggiore risoluzione i caratteri appaiono leggermente poco nitidi, e in generale si nota una certa carenza di contrasto tra lo sfondo (non molto scuro) e i caratteri. La Commodore assicura comunque che ha intenzione di distribuire una versione del PC40-III con monitor a colori, che tra l'altro dovrebbe essere lo stesso monitor destinato alla linea Amiga per far fronte alle esigenze dell'ECS.

Anche il monitor, decisamente nuovo come concezione, è influenzato dal gusto estetico inno-

vativo del computer e della tastiera: dispone infatti di una base orientabile, è di dimensioni contenute e offre alcune soluzioni originali come gli slider al posto delle consuete rotelle per la regolazione del contrasto e della luminosità. Vi è anche un interruttore nella parte anteriore, al di sotto del monitor, ma purtroppo non è stato previsto il collegamento in cascata delle alimentazioni, e il cavo dell'alimentazione non può essere separato dall'unità.

L'interno

Aprendo il PC40-III riscontriamo un piccolo particolare negativo: le dimensioni contenute del cabinet penalizzano purtroppo il numero di schede d'espansione inseribili. Infatti sulla sinistra troviamo quattro slot, tre di tipo AT e uno XT, contro gli otto di cui disponeva il PC40-40. Comunque bisogna tenere conto del fatto che in questa macchina c'è davvero tutto di serie, e quindi i quattro slot a disposizione verranno utilizzati effettivamente per espandere il sistema e non per montare accessori indispensabili come controller o espansioni RAM, mentre nel precedente modello tre slot venivano "sprecati" per una scheda controller, una scheda grafica AGA, e una scheda con interfaccia seriale e parallela.

La maggior parte dello spazio è occupato da alcune ingombranti unità che nonostante i progressi tecnologici non sono ancora state ridotte a dimensioni contenute: l'alimentatore, il disk drive e l'hard disk. Quest'ultimo è un modello slim da 3,5", che offre 40 MB e un tempo d'accesso medio di soli 28 ms. Per quanto riguarda l'alimentatore switched mode, va però detto che le notevoli dimensioni sono indice di una certa robustezza elettronica che dovrebbe garantire una corretta alimentazione della macchina in tutte le configurazioni.

Sulla scheda madre troviamo poi un vero e proprio show di tecnologia concentrata: la presenza di grossi integrati dotati di un'infinità di connessioni fa quasi venire in mente l'Amiga. Oltre alla CPU 80286, infatti, troviamo parecchi altri circuiti integrati VLSI (tra cui quello di produzione Paradise, che gestisce la grafica) e lo zoccolo per il coprocessore matematico (l'Intel 80287). Vi si trova anche l'orologio con batteria tampone e degni di nota sono inoltre gli otto chip di RAM 414256 da 256 x 4 Kbit da 100 ns di tempo d'accesso; non c'è posto per altri chip di memoria sulla scheda. È un particolare curioso che in questo computer con frequenza massima pari a 12 MHz si trovino delle RAM da 100 ns, mentre sulla scheda A2620 per l'Amiga (della quale abbiamo parlato nello scorso numero di *Commodore Gazette*), che funziona a 14,2 MHz, fossero presenti chip da 120 ns.

Un ultimo sguardo all'hard disk, prima di terminare questo esame interno. Si tratta di un modello in perfetto standard AT della Quantum, che offre prestazioni di tutto rispetto. La realizzazione della scheda dell'hard disk, così come l'intera scheda madre ci sembra ordinata e affidabile, anche se va segnalato qualche collegamento volante. Sulla scheda esiste un jumper che consente la selezione tra due possibili tipi di hard disk da collegare al controller di serie, in grado di gestire un solo hard disk: il modello Quantum o il modello Conner; tale jumper viene in aiuto all'utility SETUP nella selezione del tipo di periferica collegata.

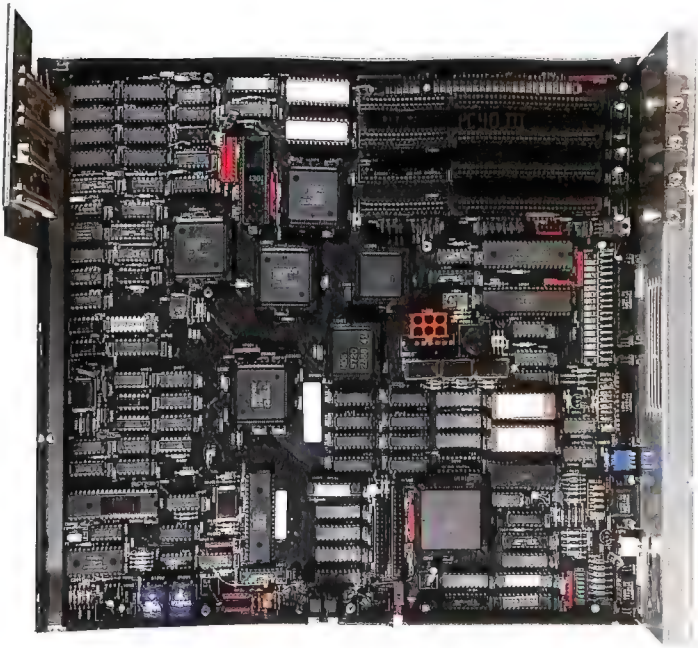
Il terreno di prova

I manuali forniti dalla casa costruttrice sono in lingua inglese, tranne quello relativo al monitor che contiene anche una sezione con le istruzioni in italiano. Il disco sistema offerto con la macchina contiene la versione 3.3 dell'MS-DOS e alcune utility necessarie per gestire le peculiarità del PC40. Viene anche fornito il Basic GW-BASIC nella versione 3.22. Due manuali sono dedicati rispettivamente al sistema operativo e al Basic: in entrambi i casi si

tratta di guide scritte in inglese. Il terzo manuale dedicato al computer spiega quali sono le peculiarità del PC40-III e contiene diverse sezioni tecniche sui modi grafici, sui clock selezionabili e così via.

Come sappiamo, la macchina è compatibile con l'IBM AT, e come tale in protected mode può indirizzare più di un megabyte di memoria (fino a 16 MB di memoria fisica, fino a 1 GB con indirizzamento virtuale), naturalmente ridotti a 640K in MS-DOS. Le frequenze di clock disponibili sono tre, selezionabili via software o attraverso combinazioni di tasti (una comodità questa non presente sui PC della serie precedente): 6 MHz (come l'AT originale), 8 e 12 MHz. La velocità di default è quella più alta, ma la macchina può essere impostata sin dalla fase di attivazione (esecuzione del file comandi AUTOEXEC.BAT) per lavorare a una frequenza diversa, magari per motivi di compatibilità con software più vecchio. Purtroppo la CPU si scontra sempre con uno stato d'attesa (wait state) per ragioni di compatibilità con il bus AT originale, e questo oscura parzialmente le prestazioni del nostro PC nei confronti di altri AT compatibili che lavorano con zero wait state.

La memoria installata di serie è di 1 MB, di cui 640K corrispondono a quelli tradizionali visti dal DOS, mentre i rimanenti 384K sono considerati memoria estesa (allocata oltre lo spazio visibile al bus degli indirizzi dell'8086, ossia oltre il primo megabyte). Questa memoria può essere disabilitata attraverso uno dei dip switch collocati sul retro del computer per collegare, per esempio, espansioni in standard LIM (Lotus, Intel, Microsoft). Senza applicazioni particolari, l'unico modo di accedere alla memoria estesa è quello di utilizzarla come RAM disk (come viene infatti configurata dal software di attivazione della macchina presente nel disco sistema fornito insieme al computer), ma questa limitazione dipende dall'MS-DOS e già diversi costruttori, come quelli appena citati, hanno cercato di porvi rimedio.



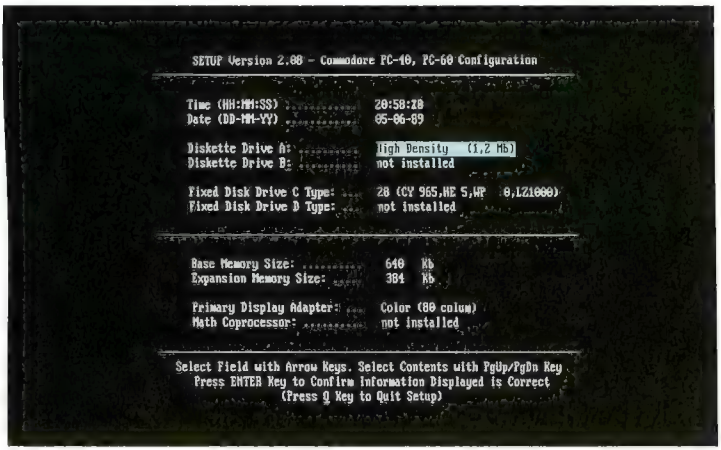
L'adattatore grafico fornito di serie è piuttosto completo in quanto può funzionare in molti modi diversi: Plantronics, Hercules, MDA, CGA, EGA per arrivare ai fasti della VGA. La circuiteria può anche essere disabilitata per cedere il controllo a una scheda video esterna. I vari modi grafici, per quanto ben noti nel mondo dei compatibili IBM, sono descritti abbastanza a fondo nell'appendice H del manuale e vengono fornite anche alcune informazioni per i programmatori. A seconda della configurazione scelta, si possono visualizzare da 40 x 25 caratteri fino a 132 x 43 in modo testo, e in grafica si parte dai classici 320 x 200 pixel del C-64 per arrivare fino a 640 x 480, ovviamente non in interlace. I colori sono sempre scelti, in numero variabile, da una palette di 262.144.

La Commodore distribuisce questo computer con l'hard disk già formattato e partizionato in due dischi logici da 20 MB. Nella prima partizione (C:) troviamo già installato l'intero contenuto del disco fornito con la macchina, ossia il sistema operativo, il GW-BASIC e le utility custom, mentre la seconda è vuota. Naturalmente, chi lo desidera può modificare lo stato delle cose creando nuove suddivisioni sul disco.

Tra i tool forniti dalla Commodore, oltre al consueto SHIPDISK che serve a parcheggiare in zona sicura le testine dell'hard disk in vista di un trasporto, troviamo alcune utility adatte a gestire la grafica, la velocità del sistema e alcuni altri parametri. VMODE è la prima di queste utility, e serve per impostare una modalità video diversa da quella automaticamente impostata all'accensione: è in pratica un analogo di MODE dedicato però al solo adattatore video. Si tenga presente che attraverso VMODE non s'imposta soltanto un modo video, ma anche un tipo

di emulazione grafica: è come se si estraesse una scheda dal computer per inserirne una diversa, ciascuna delle due con modi grafici propri. SETUP è invece un'utility per la gestione dell'intero sistema. Questa utility, tra l'altro distribuita anche con la scheda A2286 per l'Amiga (si veda il relativo articolo su questo stesso numero), si presenta con un menu interamente gestito dai tasti cursore: tale menu ci permette d'impostare l'ora e la data dell'orologio interno (il quale non viene aggiornato dal comando TIME del DOS), oppure d'impostare il tipo di memorie di massa collegate, o di segnalare la presenza di un coprocessore e così via.

Esiste poi un'altra utility di setup, residente in ROM e richiamabile con la combinazione di tasti CTRL-ALT-ESC, che svolge più o meno gli stessi compiti, evidenziando però maggiormente i tipi di hard disk previsti. Queste utility, in unione alla capacità di autoconfigurazione di cui dispone il BIOS del PC40-III, rendono molto semplice l'installazione di nuove schede o di nuove periferiche nel sistema.



L'applicazione SETUP consente d'indicare al BIOS la configurazione di cui si dispone

Il computer e le prestazioni

Parliamo ora delle prestazioni della macchina. Precisiamo subito che nei nostri test abbiamo sempre utilizzato il computer alla velocità massima, perché ci sembra un controsenso non fare un uso completo della potenza che viene messa a disposizione; solo nei casi in cui si usano programmi che presentano problemi di temporizzazione è consigliabile scegliere una frequenza di clock inferiore. A 12 MHz, questa macchina esegue dunque circa 2300 Dhrystone al secondo, il che la pone decisamente al di sopra dell'IBM AT standard, eguagliandola in pratica alla maggior parte dei compatibili provenienti dall'Estremo Oriente, e collocandola leggermente al di sotto dei 2557 Dhrystone al secondo riscontrati con il recente Compaq SLT/286, anch'esso dotato dell'Intel 80286 e funzionante a 12 MHz.

Abbiamo poi eseguito sul PC40-III una nutrita serie di test, per valutarne le prestazioni sotto tutti i profili. Manca all'appello solo la prova del coprocessore matematico, componente non fornitoci dalla casa costruttrice. Tutti i programmi sono stati compilati usando il compilatore Microsoft C versione 5.0, e si tratta di programmi volti a far eseguire al computer un certo particolare compito per *n* volte. I risultati ottenuti sul PC40-III sono poi stati confrontati con quelli relativi anzitutto a un esponente della "vecchia guardia" Commodore, ossia il PC60-40 (vedere l'articolo "I PC Commodore IBM compatibili della terza serie" pubblicato nel numero 3/1988 di *Commodore Gazette*); poi si sono raffrontati anche i risultati ottenuti con l'Amiga dotato della scheda AT Bridgeboard 2286. Per curiosità, alcuni di questi test sono stati eseguiti anche su macchine della classe XT e (con i necessari adattamenti) anche sull'Amiga funzionante nel modo natìo.

I dati ottenuti presentano un PC40-III vincente in termini assoluti sotto molti aspetti, ma soprattutto lo confermano come un'ottima macchina quanto a globalità di realizzazione, ossia per quanto riguarda l'equilibrio delle varie parti (memoria, CPU, hard disk). In pratica, tutti i benchmark di calcolo puro e di accesso alla memoria lo collocano in una ben precisa classe di prestazioni: circa il 25 per cento più lento del PC60 che, lo ricordiamo, monta un 80386 a 16 MHz. Ma chiamando in causa l'hard disk si notano prestazioni di livello superiore (come si può notare dalle tavole pubblicate) anche se i due computer dispongono di periferiche analoghe. Questa stranezza è stata poi confermata da un altro test il quale ha riportato per l'hard disk del PC60-40 un data transfer rate (velocità di trasmissione dei dati) pari a circa la metà di quello del PC40-III (239 Kbit/s contro 470 Kbit/s).

È sorprendente anche constatate che il PC60, rallentato a 8 MHz tramite l'apposita utility fornita dalla Commodore, risulta circa quattro volte più lento (anziché due) rispetto alla velocità che si ottiene con la normale frequenza di 16 MHz: questo accade solo quando si eseguono calcoli che coinvolgono la memoria, mentre quando le operazioni riguardano esclusivamente i registri interni della CPU il rallentamento del PC60 ritorna a essere quello che

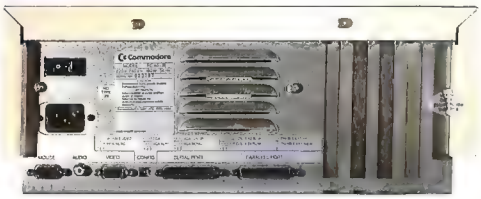
sarebbe logico aspettarsi. Di conseguenza la velocità del PC40-III a 8 MHz in quasi tutti i casi riesce a superare quella del PC60 a pari frequenza.

Questo dato ci fa supporre che il rallentamento del clock realizzato nel computer più vecchio abbia presentato dei problemi, e che rappresenti in qualche modo una forzatura rispetto al funzionamento standard. Tutto questo invece non accade nel nuovo PC40-III, dove le tre frequenze di clock disponibili equivalgono a prestazioni proporzionali.

Infine, notiamo che rispetto all'IBM AT si ottengono prestazioni del tutto identiche, sotto il profilo della potenza di calcolo, quando la frequenza di clock è la stessa cioè 6 MHz; l'hard disk si dimostra invece superiore a quelli installati sulle vecchie macchine IBM. Va però detto che alla massima frequenza di clock questo calcolatore non è al top delle prestazioni sia per quanto riguarda la pura velocità di calcolo sia per i modi video offerti: sul mercato esistono computer compatibili AT a 16 MHz (e zero wait state) e con grafica superiore a quella della scheda VGA.

Conclusioni

Questo nuovo computer Commodore dedicato all'ambiente MS-DOS appare una proposta valida per svariate esigenze professionali: non sarà adatto forse al calcolo di modelli tridimensionali di molecole, però può servire per scrivere testi, gestire fogli elettronici e archivi con una certa tranquillità. Può essere usato con le applicazioni tipiche dell'ambiente Windows, che mette a disposizione un'interfaccia grafica quasi pari a quella offerta dall'Amiga; può lavorare in Xenix 286 (lo Unix per le macchine AT), e con l'OS/2.



Per l'impiego in ambiente OS/2 bisogna in effetti considerare che a causa delle forti richieste di memoria di quell'ambiente è necessario espandere di almeno 2 MB il PC40, e questo può essere un po' costoso. Inoltre non è molto consigliabile acquistare una macchina dotata dell'80286 per usare l'ambiente MS-DOS messo a disposizione dall'OS/2 (compatibility box) a causa dell'architettura interna della CPU.

A parte queste considerazioni, possiamo dire che il prezzo è assai inferiore a quello di un IBM PS/2 50 Z (Intel 80286, clock di 10 MHz, 1 MB di RAM, hard disk da 60 MB), che costa 7 milioni e 900 mila lire, IVA esclusa. Confrontato invece con computer di altre marche, capaci di prestazioni simili, il PC40-III è più economico di un Compaq Deskpro 286/40 e di un Hewlett Packard 80286 (peraltro funzionante a 8 MHz e con un hard disk da 20 MB), mentre si rivela più elevato rispetto a prodotti della Sanyo e della Tandy.

La qualità dell'hardware è elevata e l'unico appunto si può forse muovere al monitor, che consiglieremmo di sostituire con un modello a colori al fine di apprezzare la notevole grafica VGA di cui il computer dispone.

DATI TECNICI

Microprocessore:
Intel 80286 con frequenza di clock selezionabile tra 6 MHz, 8 MHz e 12 MHz, coprocessore matematico 80287 opzionale
Memoria RAM:
1024K di serie, di cui 384K di RAM estesa su scheda madre
Monitor:
Commodore 1403 monocromatico bi-sync a fosfori bianco-carta
BIOS:
Commodore AT BIOS V2.0, pienamente AT compatibile, con capacità di autoconfigurazione a seconda delle schede installate e routine di setup incorporata
Grafica:
Adattatore video avanzato, compatibile con MDA, CGA, EGA, VGA nonché Hercules e Plantronics su scheda madre
Memorie di massa:
Un disk drive da 1,2 MB per dischi da 5,25"
Un hard disk da 40 MB, 3,5" e 28 ms di tempo d'accesso
Controller:
Controller per hard disk AT sulla scheda madre
Espandibilità:
Quattro slot di cui tre compatibili con il bus AT. Il controller può gestire un secondo disk drive o uno streamer
Dispositivi:
Controller per hard disk AT
orologio/calendario con batteria tampone
interfaccia per mouse Microsoft
porta seriale RS-232
porta parallela Centronics
switch esterni di configurazione
pulsante di reset
uscita audio
Tastiera:
Italiana, di tipo AT avanzato con 102 tasti
Dimensioni:
355 x 390 x 150 mm (LAP)
Prezzo al pubblico:
PC40-III L. 5.570.000 + IVA
Distribuzione:
Commodore Italiana spa
Viale Fulvio Testi, 280 - 20126 Milano
(Tel. 02/661231)

I RISULTATI DEI TEST

TEST	COMPUTER E FREQUENZA DI LAVORO					
	PC40-III			PC60-40		IBM AT
	6 MHz	8 MHz	12 MHz	8 MHz	16 MHz	6 MHz
TEST GLOBALI						
Dhrystone						
senza registri	1086	1562	2272	1063	2941	1086
con registri	1162	1612	2380	1162	3333	1162
Sieve						
con 100 iterazioni	69	51	33	69	26	69
Big Sieve						
con 10 iterazioni	193	143	95	185	64	n/d
TEST SPECIFICI						
Loop di conteggio						
Double 100	45	32	21	46	16	48
Long 10000	75	54	36	78	25	75
Integer 10000	38	29	19	36	15	38
Register 10000	16	12	7	9	6	15
Hard disk *(tempo d'accesso medio)*		***(28 ms)***			***(28 ms)***	***(80 ms)***
Printf 300K	27	21	20	31	12	26
Scanf 300K	36	28	18	41	21	42
Putc 300K	19	12	11	14	12	27
Getc 300K	11	11	9	12	11	12
Disk drive						
Printf 60K	25	25	25	25	24	25
Scanf 60K	23	23	23	24	24	23
Putc 60K	25	25	25	26	25	25
Getc 60K	23	22	22	22	22	23
RAM disk						
Printf 300K	23	18	11	24	8	23
Scanf 300K	39	27	19	38	13	39
Putc 300K	12	9	6	10	4	12
Getc 300K	10	6	5	9	4	10
Stampa di 1000 stringhe						
Modo testo MDA	21	18	12	36	25	14
Modo testo CGA	24	18	12	38	28	n/d
Spostamento di blocchi						
Copia stringhe (70 char) 20000	33	25	16	39	14	32
Copia strutture (160 byte) 20000	20	16	10	34	12	21
10000 chiamate a funzioni						
Senza parametri	18	13	9	22	7	18
Con tre parametri numerici	32	24	16	32	11	32

Nota i test offrono risultati espressi in secondi Numeri inferiori corrispondono a prestazioni migliori L'unica eccezione è Dhrystone maggiori Dhrystone/sec corrispondono a maggiori prestazioni

IL NOSTRO BANCO DI PROVA

Per valutare le prestazioni di una macchina esistono fondamentalmente due metodi. Il primo si basa sulla considerazione che non tutte le macchine sono uguali, ossia sul fatto che un computer può svolgere meglio compiti di calcolo puro mentre un altro può eccellere nello spostamento dei blocchi di memoria: la conclusione è che si devono scrivere routine molto particolareggiate che possano valutare nei dettagli ogni singola prestazione.

Il secondo approccio mira invece a stimare le prestazioni di una macchina in condizioni d'impiego reale, partendo dall'idea che i test troppo specifici abbiano un valore senz'altro oggettivo ma troppo teorico. Si stilano dunque alcuni programmi che svolgano calcoli "utili" o almeno che abbiano una struttura simile a quella dei programmi reali.

Da parte nostra, abbiamo deciso di saminare la macchina eseguendo tutti e due i tipi di test, per trarre conclusioni più complete. Come test di tipo generale (di "alto livello") abbiamo utilizzato i famosi Dhrystone e Sieve. Il primo è basato su un insieme di operazioni che rispecchia l'attività media di un programma reale, mentre il secondo, pur impegnando parecchie risorse della macchina, svolge soprattutto un compito "tangibile" ossia la ricerca dei numeri primi con il metodo del crivello di Eratostene. Il test Dhrystone offre un risultato espresso nell'unità di misura Dhrystone/sec, e presenta il vantaggio di essere un algoritmo standard la cui validità è riconosciuta universalmente. Il risultato che restituisce può essere influenzato solo dal tipo di compilatore impiegato.

Per quanto riguarda i test di basso livello ci siamo invece affidati a routine scritte in C, linguaggio che consente un controllo abbastanza diretto del calcolatore pur essendo trasferibile da una macchina all'altra. Osservando il listato si può notare che in ciascun loop, la cui durata viene decisa tramite un parametro della linea comando, vengono eseguiti compiti molto specifici. Questi test separano e analizzano ciò che il test Dhrystone sintetizza in un risultato unico.

Vi sono sei tipi di calcoli tra variabili numeriche, definite rispettivamente come *long* (4 byte), *integer* (2 byte), *register* (2 byte ma contenute in un registro della CPU), *char* (1 byte), *float* (4 byte in virgola mobile) e *double* (8 byte in virgola mobile). I risultati dei calcoli con char e float sono stati tralasciati dal momento che erano rispettivamente uguali a quelli ottenuti con integer e double.

I test riguardanti le variabili allocate in memoria non si differenziano molto l'uno dall'altro, nel senso che utilizzano nello stesso modo e nella medesima percentuale le risorse del computer: semplicemente, mutano la complessità di calcolo e il tipo d'istruzioni della CPU coinvolte. Nel caso delle variabili in virgola mobile si arriva anche a controllare l'efficacia dell'emulazione software del coprocessore matematico, la quale è tuttavia peculiare del compilatore (che la mette a disposizione nelle librerie linked) e non del calcolatore.

Quindi eseguendo questi test su macchine dotate delle stessa CPU, i fattori chiamati in causa sono la frequenza di clock e il tipo di memoria (velocità, wait state). Invece, esaminando i valori forniti dal loop che usa i registri della CPU, ci svincoliamo dalla memoria (tranne ovviamente per quel che riguarda la lettura delle istruzioni di cui è composto il programma) e misuriamo la pura velocità di calcolo del microprocessore. Facendo un confronto possiamo ricavare informazioni anche sulla memoria.

Seguono poi quattro test per la scrittura e la lettura dei file su tutti i dispositivi, suddivisi a seconda del tipo di funzione usata per trasferire i singoli caratteri: due fanno uso delle funzioni di I/O putc e getc, altre due usano le funzioni standard di libreria printf e scanf, le quali agiscono su stringhe di caratteri: in questi ultimi due casi si nota un tempo generalmente superiore in quanto vengono effettuati anche alcuni calcoli. Troviamo poi un banale loop di scrittura a video, e infine alcuni test in cui vengono svolte operazioni di trasferimento di dati, sia con loop espliciti sia mediante le istruzioni di carica-e-incrementa tipiche dell'80x86 (REP STOS, REP LOADS ecc.); un'altra sezione controlla l'overhead che si ottiene richiamando una funzione con e senza argomenti.

Si tenga presente che l'unico modo per essere sicuri che il codice macchina risultante faccia effettivamente il lavoro da noi desiderato consiste nell'analizzare (disassemblare) il codice prodotto dal compilatore, ed è così che ci siamo comportati.

Listato: *Benchmark*

```
#include <stdio.h>
#include <time.h>
#include <stdlib.h>
#include <malloc.h>
#include <ctype.h>

char *nomefile="pippo";
char buffer[20];
#define ITERS    16383
#define BIGITERS 163830

void sieve(unsigned int limite) {
   unsigned int i;
   unsigned totale,k;
   char *sieve;
   if(!(sieve=calloc(ITERS+1,sizeof(char)))) {
      puts("Memoria insufficiente");
      exit(99);
      }
   printf("Sieve: %u iterazion%c\n" limite,limite
         == 1 ? 'e' : 'i');
   while(limite--) {
      totale=0;
      for(i=2; i <= ITERS; i++) {
         if(!sieve[i]) {
            for(k=i*2; k<=ITERS; k+=i) sieve[k]=1;
            totale++;
            /* printf("Numero primo = %d\n",i) */
            }
         }
      }
   printf("Numeri primi trovati  1  %u = %u\n"
         ITERS totale);
   free(sieve);


void bigsieve(unsigned int limite)
   unsigned long i;
   unsigned long totale,k;
   char huge *sieve;
   if(!(sieve=halloc(BIGITERS+1L,sizeof(char)))) {
      puts("Memoria insufficiente");
      exit(99);
```

```
    }
  printf("Big Sieve: %ld iterazion%c\n",limite,
         limite== 1 ? 'e' : 'i');
  while(limite--) {
     for(i=0; i <= BIGITERS; i++) sieve[i]=0;
     totale=0;
     for(i=2; i <= BIGITERS; i++) {
        if(!sieve[i]) {
           for(k=i*2; k<=BIGITERS; k+=i)
             sieve[k]=1;
           totale++;
        /* printf("Numero primo = %ld\n",i); */
           }
        }
     }
  printf("Numeri primi trovati [1..%ld] = %ld\n",
         BIGITERS,totale);
  hfree(sieve);
  }

void ciclo(unsigned int limite) {
   unsigned long i,l;
   printf("Long: %u iterazioni\n",limite);
   l=250500L;
   while(limite--) for(i=250000L; i<l; i++);
   }

void ciclo0(unsigned int limite) {
   unsigned char i,l;
   printf("Char: %u iterazioni\n",limite);
   l=253;
   while(limite--) {
      for(i=3; i<l; i++);
      for(i=3; i<l; i++);
      }
   }

void ciclo1(unsigned int limite) {
   register unsigned i,l;
   printf("Register: %u iterazioni\n",limite);
   l=25500;
   while(limite--) for(i=25000; i<l; i++);
   }

void ciclo2(unsigned int limite) {
   unsigned i,l;
   printf("Integer: %u iterazioni\n",limite);
   l=25500;
   while(limite--) for(i=25000; i<l; i++);
   }

void ciclo3(unsigned int limite) {
   double i,l;
   printf("Double: %u iterazioni\n",limite);
   l=750.0;
   while(limite--) for(i=250.0; i<l; i++);
   }

void ciclo4(unsigned int limite) {
   float i,l;
   printf("Float: %u iterazioni\n",limite);
   l=750.0;
   while(limite--) for(i=250.0; i<l; i++);
   }

void stampa(unsigned long limite) {
   while(limite--)
     printf("Il valore e' %ld ",limite);
   puts("");
   }

void NoOpen() {
   printf("Non posso aprire il file %s\n",buffer);
   exit(99);
   }

void scrivi(unsigned long limite) {
   FILE *f1;
   printf("Printf [%s]: %ld iterazioni\n",buffer,
          limite);
   if(f1=fopen(buffer,"w")) {
      while(limite--) fprintf(f1,"pippo ");
      fclose(f1);
      }
   else NoOpen();
   }

void leggi(unsigned long limite) {
   FILE *f1;
   char string[10];
   printf("Scanf [%s]: %ld iterazioni\n",buffer,
          limite);
   if(f1=fopen(buffer,"r")) {
      while(limite--) fscanf(f1,"%s",string);
      fclose(f1);
      }
   else NoOpen();
   }

void scrivi1(unsigned long limite) {
   FILE *f1;
   printf("Putc [%s]: %ld iterazioni\n",buffer,
          limite);
   if(f1=fopen(buffer,"w")) {
      while(limite--) {
         putc('p',f1);
         putc('i',f1);
         putc('p',f1);
         putc('p',f1);
         putc('o',f1);
         putc(' ',f1);
         }
      fclose(f1);
      }
   else NoOpen();
   }

void leggi1(unsigned long limite) {
   FILE *f1;
   printf("Getc [%s]: %ld iterazioni\n",buffer,
          limite);
   if(f1=fopen(buffer,"r")) {
      while(limite--) {
         getc(f1);
         getc(f1);
         getc(f1);
         getc(f1);
         getc(f1);
         getc(f1);
         }
      fclose(f1);
      }
   else NoOpen();
   }

void copiastrs(unsigned int limite) {
   char *part="Stringa numero uno per il test
 sullo spostamento di blocchi";
   char dest[100];
   int z;
   printf("CopiaStringhe70: %u iterazioni\n",
          limite);
   while(limite--)
      for(z=0; z<10; z++)
         strcpy(dest,part);
   }

void copiastructs(unsigned int limite) {
   struct {
      char a[140];
```

```
      float b[5];
      } a,b;
   int z;
   printf("CopiaStrutture160: %u iterazioni\n",
          limite);
   while(limite--)
      for(z=0; z<10; z++)
         a=b;
   }

void dummy() {
   dummy1();
   }

void dummy1() { }

void calls(unsigned int limite) {
   int z;
   printf("Funzioni: %u iterazioni\n",limite);
   while(limite--)
      for(z=0; z<25; z++) {
         dummy();
         dummy();
         }
   }

void dummarg(int a,long b, float c) {
   dummiarg(a,b,c);
   }

dummiarg(int a,long b ,float c) {
   return a;
   }

void callargs(unsigned int limite) {
   int z;
   printf("FunzioniCon3Argomenti: %u iterazioni\n",
          limite);
   while(limite--)
      for(z=0; z<10; z++)
         dummy(10,200000L,30.0);
   }

main(int argc, char **argv) {
   unsigned long i;
   long ora,finora;
   int t;
   i=atol(argv[2]);
   if(argc==1 || !isdigit(*argv[1])) {
    printf("Test v1.3, (C) G.DAR  1989,1990 ");
    printf("-  Versione MS-DOS\n\n");
    printf("Uso: TEST <numero> <limite iter.>");
    printf(" [dispositivo]\n");
    puts("dove numero:");
    puts(" 1. Sieve");
    puts(" 2. Big Sieve");
    puts(" 3. Scrive su file stringhe da 6 char");
    puts(" 4. Legge da file stringhe da 6 char");
    puts(" 5. Scrive su file sequenze di 6 char");
    puts(" 6. Legge da file sequenze di 6 char");
    puts(" 7. Loop con variabili char");
    puts(" 8. Loop con variabili register");
    puts(" 9. Loop con variabili integer");
    puts("10. Loop con variabili long");
    puts("11. Loop con variabili float");
    puts("12. Loop con variabili double");
    puts("13. Stampa stringhe di char a stdout");
    puts("14. Copia stringhe da 70 caratteri");
    puts("15. Copia strutture da 160 byte");
    puts("16. Esegue chiamate di funzioni");
    printf("17. Esegue chiamate di funzioni con");
    printf(" 3 argomenti misti e ritorno\n");
    printf("e limite iter. <= 4.294.967.295 nei ");
   puts("casi 3,4,5,6: 65535 negli altri casi");
    exit(1);
    }
   else if(argc < 3) exit(99);
   t=atoi(argv[1]);
   if((t > 2) && (t<7)) {
      if(argc==4) {
         strcpy(buffer,argv[3]);
         }
      strcat(buffer,nomefile);
      }
   if(!i || (((t<3) || (t>6)) && i>65535L)) {
      puts("Valore illegale");
      exit(99);
      }
   printf("Tempo iniziale : %9ld ticks\n",
          time(&ora));
   switch(t) {
      case 1:
         sieve(i);
         break;
      case 2:
         bigsieve(i);
         break;
      case 3:
         scrivi(i);
         break;
      case 4:
         leggi(i);
         break;
      case 5:
         scrivi1(i);
         break;
      case 6:
         leggi1(i);
         break;
      case 7:
         ciclo0(i);
         break;
      case 8:
         ciclo1(i);
         break;
      case 9:
         ciclo2(i);
         break;
      case 10:
         ciclo(i);
         break;
      case 11:
         ciclo4(i);
         break;
      case 12:
         ciclo3(i);
         break;
      case 13:
         stampa(i);
         break;
      case 14:
         copiastrs(i);
         break;
      case 15:
         copiastructs(i);
         break;
      case 16:
         calls(i);
         break;
      case 17:
         callargs(i);
         break;
      }
   printf("Tempo finale   : %9ld ticks\n",
          time(NULL))
   finora=time(NULL)-ora
   printf("Tempo impiegato: %9ld second%c\n",
          finora,finora==1?'o':'i')
   exit(0);
   }
```

■

LA RISCOSSA DEL SOFTWARE MADE IN ITALY

Crimetown Depths è già un mito per chi ha potuto vederne le demo: Alessandro Tento, uno dei fondatori della software house italiana E-motion, ci parla del programma e dei progetti futuri

di Fabio Rossi

Con quella faccia un po' così,
quell'espressione un po' così,
che abbiamo noi prima di andare a Genova...
ed ogni volta ci chiediamo
se quel posto dove andiamo
non ci inghiotta e non torniamo più

È una strana coincidenza sentire Paolo Conte dall'autoradio proprio mentre imbocco la sinuosa sopraelevata che dall'autostrada scende in città. In effetti, sotto questo cielo incerto, Genova ha un aspetto abbastanza inquietante con le sue strade strette e tutte in pendenza. Da qualche parte lungo i suoi meandri dove è impossibile trovare un parcheggio c'è la sede della E-motion Software, la software house italiana più chiacchierata degli ultimi mesi: le riviste specializzate in videogiochi fanno a gara per pubblicare le foto della loro demo di *Crimetown Depths*, un videogame per l'Amiga graficamente superbo che si dice abbia anche una struttura eccezionalmente complessa.

Riesco miracolosamente a trovare un parcheggio vicinissimo alla sede della E-motion, dove mi attende Alessandro Tento, uno dei responsabili della sezione grafica, un ragazzo molto giovane e fisicamente più simile a un atleta che a un programmatore che passa le sue ore davanti a un monitor. Ci incrociamo nell'androne del palazzo senza riconoscerci e solo in seguito ci accorgiamo della gaffe. L'imbarazzo fortunatamente passa subito, e in breve tempo chiacchieriamo come vecchi amici.

L'appuntamento organizzato solo otto ore prima non gli ha dato il tempo di riordinare l'ufficio abbandonato in fretta e furia dai suoi tre colleghi partiti per l'Inghilterra, per partecipare al Commodore Show. Ci spostiamo così nel suo appartamento, dove finalmente posso far partire il registratore e cominciare con le domande...

D. *Iniziamo proprio dal principio: com'è nata la E-motion Software?*

R. Da un'idea mia e di Marino Roberto, il titolare della Newtronic di Genova. Tornavo da un lungo viaggio a Londra, dove avevo allacciato qualche contatto, e poiché sapevo che entrambi avevamo il sogno nel cassetto di riuscire a vendere un gioco per l'Amiga, mi sono rivolto a lui per realizzare l'idea. Era il gennaio 1988, e in quel periodo ho realizzato il primo logo della ditta, un ghepardo che corre, che appariva

Una schermata introduttiva del gioco Crimetown Depths

anche in una demo animata.

Da allora al maggio '88 abbiamo perso un sacco di tempo a causa di alcuni programmatori che ci avevano contattato promettendo scintille e saette ma poi si sono rivelati un bluff. Le nostre esperienze sono aumentate di mese in mese e abbiamo imparato a valutare le capacità altrui riuscendo a trovare Max (Massimo Magnasciutti), Paolo Costabel e Roberto Sagoleo.

D. *Come vi siete divisi i compiti?*

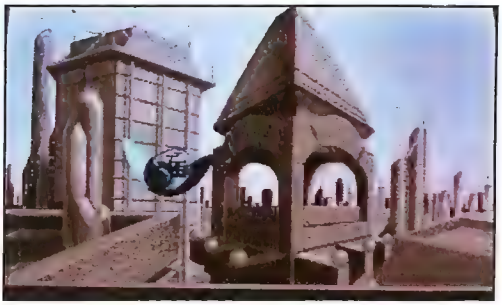
R. Io, che mi ero comprato l'Amiga soprattutto per disegnare e ho molta esperienza nel disegno con l'aerografo, mi occupo della parte grafica. Max è un grafico ma anche un vulcano d'idee, e la cosa più difficile con lui è proprio fermarlo: di giorno lavora e di notte dorme e sogna nuove idee da realizzare. Paolo Costabel si occupa della programmazione di tutte le routine insieme a Roberto Sagoleo, usando la loro grande esperienza col C. Sono veramente molto bravi: Roberto è passato all'Amiga dal C-64 solo da un anno, e già conosce a menadito i quattro manualoni della Commodore.

D. *E Marino Roberto?*

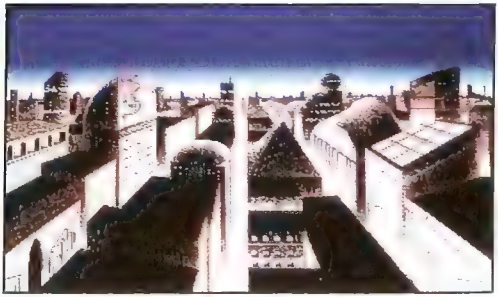
R. Si occupa della parte finanziaria. È una persona molto disponibile e disinteressata, ed ero in contatto con lui già prima che nascesse il progetto: ha fornito i capitali e ci finanzia per la soddisfazione di vedere una ditta italiana raggiungere un successo riconosciuto a livello europeo. Abbiamo addirittura occupato una parte del suo ufficio. Il suo contributo è fondamentale.

D. *Un sacco di gente è rimasta affascinata dalle demo di* Crimetown Depths. *Cosa ci puoi raccontare di questo gioco misterioso?*

R. Avevamo un sacco d'idee, all'inizio: simulazioni, giochi con motociclette e tante altre cose. Alla fine è prevalso il filone fantascientifico che è molto in auge di questi tempi. La sceneggiatura è nata in collaborazione, ma io avevo ideato qualcosa di molto diverso e ci sono stati alcuni disaccordi fra Max e me. Poi, dato che i programmatori erano solo due, si sono dovuti dedicare entrambi allo stesso progetto. L'idea iniziale era di creare uno shoot'em up classico con una bella grafica, tanto per saggiare il mercato. Ma abbiamo continuato ad aggiungere un elemento dopo l'altro sino a raggiungere dimensioni enormi, tali che negli ultimi tempi avevamo delle difficoltà a riordinare tutti gli elementi. Avevamo in mente un piccolo romanzo di fantascienza e a poco a poco ci siamo ritrovati con una specie di enciclopedia, grazie anche agli incoraggiamenti della Mirrorsoft, che ci hanno galvanizzati.



Nel cielo di Crimetown si svolge una delle sezioni shoot'em up



Una panoramica a volo d'uccello della "città del crimine"

D. *Quali sono i vostri rapporti con la Mirrorsoft?*

R. Quando abbiamo cominciato avevamo solo due alternative: creare e commercializzare da soli le nostre cose in Italia, oppure trovare un produttore/distributore che curasse i nostri interessi in tutta Europa. Abbiamo spedito alcune demo alle case più famose sperando in qualche reazione. Con nostra grande sorpresa dopo neanche sette giorni è arrivata una telefonata della Mirrorsoft, che abbiamo accolto con veri salti di gioia. Dopo tre o quattro mesi, con rapporti frammentari a causa della distanza e continui incoraggiamenti a ingrandire e migliorare *Crimetown*, abbiamo firmato un contratto per due giochi.

(Il contratto è incorniciato, appeso a una parete proprio sopra una montagna di cartelle da disegno piene di talloncini "Air Mail").

D. *Tu ti occupi della parte grafica dei giochi: quali strumenti utilizzate nella creazione degli scenari e delle vostre spettacolari animazioni?*

R. Sono un maniaco di *Deluxe Paint*, che possiedo in tutte e tre le versioni, e non uso praticamen-

te nessun'altra utility commerciale. Costabel ha realizzato alcune utility fra cui *BOBDrainer* per l'animazione dei BOB, e *Dedalus*, un editor per labirinti "alla *Barbarian*" che abbiamo utilizzato per *Crimetown Depths*. *Dedalus* gestisce sino a 256 moduli grafici realizzati con *Deluxe Paint*, che vengono assemblati su una griglia e possono essere spostati.

Una piccolo trucco è quello di realizzare modellini in plastilina e digitalizzarli in tutte le inclinazioni possibili per ottenere animazioni più realistiche.

D. *E in campo audio? Non mi sembra di aver sentito nominare nessun tecnico del suono.*

R. Infatti non ne abbiamo. Per la prima demo ci siamo avvalsi di Massimo Bracco, che è un compositore di professione molto bravo ma troppo impegnato per dedicarsi a questa attività. A parte la realizzazione di piccoli brani originali da campionare, la cosa migliore è avere una grande libreria di suoni da miscelare. Campioniamo continuamente suoni tratti da un sacco di fonti: compact disc, dischi di effetti speciali, film o rumori che realizziamo noi stessi nei modi più diversi. Le possibilità non mancano!

D. *Un'ultima domanda "tecnica": quali macchine utilizzate?*

R. Lavoriamo con un Amiga 1000, un 2000 e tre 500. Abbiamo inoltre programmatori esterni che si occupano delle conversioni per Atari, C-64 e PC IBM. Trovare programmatori in gamba per certe macchine, escluso il C-64, è ancora più difficile che per l'Amiga. I nostri sono tutti universitari con molta esperienza.

D. *Vicino al tuo computer vedo pochissimi giochi: ci sono* TV Sports Football, Roger Rabbit *e poco d'altro. Non vi siete ispirati a nessun altro lavoro?*

R. Ci interessano solo alcune soluzioni tecniche particolarmente buone, come ad esempio certe mascherature, ma niente in campo grafico. La maggior parte dei prodotti esteri è graficamente molto povera, e la struttura è anche peggiore.

D. *Non ti sembra di essere troppo critico nei confronti della concorrenza?*

R. I giochi mi stufano tutti, tranne forse i prodotti della Psygnosis (pronuncia il nome "all'inglese", con le vocali lunghissime, e si mette a ridere) e i primi Cinemaware, che però hanno vita troppo limitata. Ammiriamo molto i Bitmap Brothers, i nostri colleghi-concorrenti.

E-motion significa emozione ma anche movimento: noi cerchiamo di fare giochi che propongano una grafica simile a quella Cinemaware, che è molto apprezzata negli Stati Uniti, e una struttura estremamente movimentata che si adegui al mercato europeo ed eventualmente a quello giapponese, dove l'utenza è nettamente più giovane.

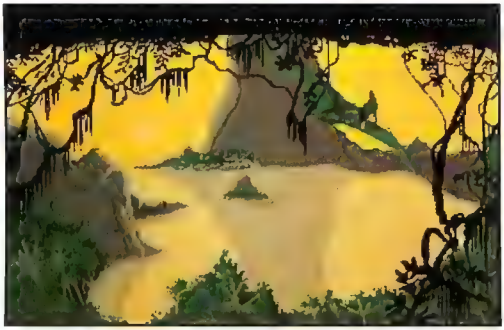
D. *Il contratto con la Mirrorsoft parla di due giochi. Quale sarà il secondo?*

R. Il titolo di lavorazione è *The Last Temple*. Non è granché, ma confido in un'ispirazione futura.

La storia è ambientata in un'epoca indefinita. Il personaggio è un monaco guerriero che secondo una leggenda deve partire per un viaggio alla ricerca di una principessa con cui accoppiarsi perché la razza dei monaci possa avere una discendenza. Il titolo si riferisce al fatto che bisogna raggiungere sette templi, ognuno con un guardiano da sconfiggere.

Ogni fase è riferita a un paesaggio ben definito: si partirà da una zona montuosa che rappresenta la Terra, poi ci saranno Acqua, Aria, Fuoco, Uomo, Fauna e un livello finale misterioso. Si dovrà volare a bordo di una creatura chiamata "Camaleosauro" o "Camaleopteryx", che è un incrocio

continua a pagina 60



Studio preparatorio per il primo quadro di The Last Temple

Primo piano del feroce protagonista di The Last Temple

CRIMETOWN DEPTHS

Le due parole che formano il titolo dell'opera prima della E-motion Software sono sulla bocca degli utenti più informati da oltre un anno, ma a tutt'oggi nessuno era riuscito a sollevare l'alone di mistero che ne circondava la trama e la struttura. L'intervista con Alessandro Tento è stata un'ottima occasione per indagare sugli aspetti meno tecnici del programma, che qui vengono svelati per la prima volta.

Crimetown Depths è ambientato su di un lontano pianeta governato da un onnipotente imperatore: fra le sue megalopoli metalliche, una è diventata col tempo un covo di malviventi, assurgendo al rango di capitale del crimine.

Nessun intervento ufficiale è riuscito a liberare le strade di quella che è stata ormai ribattezzata "Crimetown", e il continuo degenerare della situazione ha spinto l'imperatore a convocare i più famosi mercenari in circolazione, cui viene delegata l'opera di ripulire la città. Il giocatore interpreta il ruolo di uno di questi giustizieri, impegnato a raggiungere i livelli più profondi della città, dove si annidano le fonti della corruzione.

L'ambiente a sviluppo verticale di Crimetown nasconde innumerevoli insidie, costituite sia dall'eterogenea popolazione aliena dei suoi vicoli metallici sia dalle pericolanti strutture high-tech il cui stato di abbandono potrebbe risultare letale. Per affrontare le varie insidie, il nostro personaggio indossa uno scafandro corazzato dotato di uno zaino a razzo che permette balzi prodigiosi. Per le numerose azioni di forza, dispone inoltre di un vero e proprio arsenale di potentissime armi, che possono però essere trasportate solo una alla volta.

I labirintici cunicoli della città del crimine non nascondono solo insidie, comunque: fra mostri mutanti e ascensori a levitazione magnetica troviamo infatti negozi e locali che possono fornire armi aggiuntive o indizi preziosi per il proseguimento della missione.

Nonostante la massiccia presenza di scenari dalla geografia contorta, vagamente simili a quelli visti in *Obliterator*, *Crimetown Depths* non è solo un grosso labirinto digitale. Nelle varie sottosezioni che lo compongono, il giocatore deve affrontare sequenze di shoot'em up tridimensionali e scoprire il funzionamento delle camere di teletrasporto, solo per citare alcune delle situazioni che abbiamo incontrato nel corso di una breve sessione dimostrativa.

Il finale del gioco, che a detta dei suoi realizzatori sarà raggiungibile solo dopo parecchie ore di studio e di azione, è rigorosamente top secret e non ci è stato rivelato nemmeno parzialmente: tutto quello che sappiamo è che si tratterà di una sorpresa che prolungherà ulteriormente la vita del programma.

Se la trama di *Crimetown Depths* non è particolarmente originale, non si può certo dire lo stesso della sua realizzazione tecnica. La cosa che stupisce di più è l'assoluta unicità dello stile grafico, che mette in luce la profondità delle ricerche effettuate dai suoi realizzatori. Le animazioni, gli scroll parallattici e le notevoli tecniche di mascheratura utilizzate nelle sequenze tridimensionali sono di ottima qualità, e appaiono ancor più sorprendenti quando si considera la relativa inesperienza nel campo dei videogiochi dei giovani programmatori della E-motion Software.

Consigliamo a coloro che stanno già fremendo per l'impazienza di mettere le mani su una copia del programma di rilassarsi: la tormentata gestazione del gioco non sembra destinata a concludersi prima del Natale 1989, e anche questo termine potrà essere soggetto a qualche slittamento. A titolo di consolazione possiamo soltanto assicurarvi che la versione definitiva del programma, che apparirà sotto l'egida della Imageworks su un minimo di due dischi, sarà recensita in anteprima sulle pagine di *Commodore Gazette* non appena i suoi autori metteranno la parola fine al lavoro.

F.R.

di animali ispirati ai disegni di Moebius (quelli della sigla di *Mixer*) e di Frazetta. Per la grafica ci serviremo della tecnica del rotoscopio utilizzata in *Dragon's Lair*, ma non in modo così limitante: i risultati grafici saranno gli stessi, ma il giocatore potrà muoversi dove vuole e fare quel che vuole. Useremo anche alcune routine ideate da Paolo Costabel che gestiscono una grafica a vettore tanto elaborata da tendere a quella frattale.

Come in *Crimetown Depths* ci sarà una sorpresa finale, perché il nostro obiettivo è quello di stupire i giocatori. Per superare ogni quadro bisognerà trovare una strategia particolare, e ci sarà molto da pensare perché non lasceremo nulla al caso. Non vogliamo fare come in certi giochi dove all'improvviso si trovano spade appese agli alberi o altri elementi assolutamente illogici.

D. *Quali sono i progetti a lungo termine della E-motion Software?*

R. La nostra idea è di accumulare abbastanza capitali da poter acquistare un paio di workstation grafiche e darci alla pubblicità, continuando comunque a occuparci di giochi.

Per quanto riguarda il settore dei giochi abbiamo idee molto vaghe, ma ho fatto degli studi per un remake con alcuni aggiornamenti di *Viaggio allucinante*, un film basato su un racconto di Asimov. Ci sarà un veicolo singolo che dovrà salvare una serie di pazienti colpiti da diverse malattie. Dovrebbe avere una visione in soggettiva e non sarà come *The Surgeon*, dove per arrivare alla fine si doveva essere almeno laureati in medicina. Sarà molto più intuitivo.

D. *Si può vivere facendo giochi?*

R. Sì! I computer venderanno sempre di più: anche se potrà esserci qualche flessione occasionale, il mercato continuerà a crescere, e il nostro contratto comprende anche eventuali conversioni per console o giochi da bar.

D. *C'è qualcosa in particolare che vuoi dire della E-motion Software?*

R. Be'... (lunga pausa). Operiamo in Italia ma non per l'Italia perché il mercato è troppo giovane e troppo affollato di pirati. Quando all'estero scoprono che sono italiano, la prima cosa che mi dicono è: «Da voi ci sono degli ottimi "crackers"!».

Ma gli italiani sanno fare ben altro, e noi vogliamo dimostrarlo. È come avere per le mani un fantastico motore ad acqua, e non avere una carrozzeria: ci sono grandi potenzialità da sfruttare. Il nostro obiettivo è evolverci... ed essere nell'elenco delle prime cinque migliori software house entro tre/quattro anni.

A causa di un impegno di Alessandro siamo costretti a interrompere qui l'intervista. Ma riesco a passare ancora un paio di minuti ad ammirare alcune schermate di studio dei prossimi giochi della E-motion Software, che mi vengono spiegate a raffica. In ogni disegno salta all'occhio quella cura per i particolari che ha reso famoso *Crimetown Depths* prima ancora della sua pubblicazione. Mentre mi allontano dalla sede della software house, ho la sensazione di essere stato uno dei primi testimoni della nascita di qualcosa di eccezionale, che farà parlare molto di sé in futuro.

Dopo un lungo peregrinare sotto un cielo sempre indeciso tra la pioggia e il sereno, ritrovo la mia auto bloccata su tutti i lati da macchine parcheggiate persino in tripla fila. Torno a pensare alle parole di Conte: vuoi vedere che aveva ragione lui? ■

LA IHT GRUPPO EDITORIALE CERCA

▷ **Redattori tecnici per la propria redazione** ◁
La IHT cerca un redattore da inserire nei quadri del personale. La località di lavoro è Milano.
Inviare curriculum dettagliato.

▷ **Collaboratori esterni** ◁
Commodore Gazette cerca nuovi articolisti
Inviare articoli saggio e curriculum.

▷ **Agenti pubblicitari** ◁
Per la testata Commodore Gazette la IHT cerca agenti pubblicitari
o agenzie di vendita degli spazi pubblicitari.

▷ **Traduttori** ◁
La Divisione Libri cerca traduttori ai quali affidare la traduzione dall'inglese all'italiano
di testi tecnici sull'Amiga. Inviare curriculum dettagliato.

Gli interessati possono scrivere a:
IHT GRUPPO EDITORIALE – Via Monte Napoleone, 9 – 20121 Milano

UNA COMMODORE PICCINA PICCINA AL COMDEX DI CHICAGO

Tutta tesa a sfondare nel mercato professionale, la Commodore ha fatto una magra figura proprio alla più importante fiera dedicata agli operatori del settore. Quando imparerà?

di Ervin Bobo

Ospitato da due dei tre padiglioni che compongono il McCormick Center di Chicago, il Comdex (10-13 aprile '89) si presenta subito come la più importante mostra professionale dedicata ai computer di tutto il Nordamerica. Per fare un paragone, il Consumer Electronics Show occupa tutti e tre i padiglioni, ma i computer e i loro accessori devono condividere lo spazio disponibile con attrezzature stereo e video, forni a microonde, orologi digitali, telefoni e televisori.

Il Comdex invece è riservato soltanto ai computer e al software, e nell'ambiente professionale i computer sono praticamente tutti PC IBM o compatibili, ad eccezione del Macintosh a cui era dedicata una sezione più piccola, ribattezzata col nome di "Macdex".

A differenza del Consumer Electronics Show, il Comdex è aperto soltanto agli operatori commerciali, ovvero fabbricanti, distributori, rivenditori e ovviamente i responsabili degli acquisti delle varie aziende. Tutto sembra puntare sui PC e sul loro uso professionale. E la Commodore?

Alla Commodore non resta che starsene miseramente seminascosta in fondo a una delle sale principali, a fare timidi sforzi per mantenere vivo il proprio nome all'interno della comunità degli affari.

È evidente che il paragrafo precedente esclude subito questo pezzo dallo scontato ottimismo dei soliti réportage che vengono pubblicati dopo le varie mostre, e lo colloca invece tra gli articoli di protesta. Ma crediamo che sia questa la direzione giusta, perché in quanto utenti Commodore abbiamo un interesse profondo e legittimo in ciò che la Commodore, come entità aziendale, fa o non fa per assicurare la favorevole accoglienza e la reale sopravvivenza dei prodotti in cui riponiamo il nostro interesse.

Lo stand Commodore, in disparte rispetto alle zone più animate del Comdex

Facciamo un passo indietro. Da qualche tempo, la Commodore sta concentrando i suoi sforzi più sul Comdex che sul Consumer Electronics Show con l'eccezione

del gennaio 1987 quando si presentò in forze al CES di Las Vegas.

In quell'occasione era stato approntato uno stand a due piani al centro del salone, circondato da una serie di tavoli sui quali troneggiavano parecchi C-64, il prototipo del C-128D (che per gli USA rappresentava una novità, n.d.r.) e moltissimi Amiga. Le varie macchine non erano manovrate da personale della Commodore, ma erano affidate a rappresentanti dei maggiori editori di software, ciascuno dei quali dava dimostrazione delle sue più recenti e più straordinarie realizzazioni.

Mentre osservavo i capannelli di persone riunite attorno a ciascuna macchina, ho potuto notare che nel padiglione centrale i dirigenti esecutivi della Commodore venuti da tutto il mondo intrattenevano giornalisti, proprietari di catene di negozi e grossisti, cercando di stringere i giusti contatti: tutto era come doveva essere in una mostra commerciale ben riuscita.

computer. In due anni c'era da aspettarsi che qualcosa avessero imparato... ma se è stato così nessuno se n'è accorto.

Benché lo stand fosse piuttosto grande, e attrezzato con un'insegna che l'avrebbe messo immediatamente in evidenza in una mostra di livello inferiore, al Comdex si faceva addirittura fatica a trovarlo. A una mostra come questa, i partecipanti di maggiore spicco si dovrebbero poter individuare con una semplice occhiata alle insegne che campeggiano sulla fila di stand... invece per trovare la Commodore era indispensabile fare riferimento alla piantina e controllare ogni tanto i numeri che contrassegnavano le corsie.

sezione riservata al desktop publishing, la Gold Disk illustrava il suo programma *Professional Page*. E così via, tutt'attorno allo stand centrale. Per ogni macchina c'erano i rappresentanti di qualche software house che descrivevano e mostravano *un solo* prodotto: il prodotto che meglio si adattava, tra quelli disponibili, a illustrare quanto efficientemente l'Amiga era in grado di soddisfare i requisiti di un determinato settore.

Il risultato? I vari display erano monotoni, poco interessanti e praticamente ignorati. Uno schermo "multi-media" su una parete, composto di circa 25 monitor che mostravano tutti le stesse immagini, riusciva ad attirare l'attenzione di una decina di persone, mentre le "stazioni di lavoro" ne tenevano occupate altre due o tre. Se qualche dirigente della Commodore ha partecipato alla mostra, noi non ce ne siamo accorti; le sole persone della Commodore che si sono viste sono state una gentile signora che lavora nel reparto grafico della rivista *Commodore Magazine* (che sembrava essere la responsabile dell'organizzazione) e la signora che dirige il settore account della Fleishman-Hilliard (bravissima a evitare ogni domanda).

Scarsa la presenza dei dirigenti Commodore al McCormick Center di Chicago

Il Comdex

Con la sua partecipazione al Comdex '89, la Commodore ha confermato il trend negativo nelle relazioni pubbliche cominciato da quando l'azienda si è messa nelle mani dell'agenzia Fleishman-Hilliard. Anche se è conosciuta e rispettata negli Stati Uniti come una delle migliori agenzie di pubbliche relazioni (e anche se uno dei fondatori dell'agenzia è uno dei miei migliori amici) la Fleishman-Hilliard ha commesso l'errore di affidare la Commodore a un'équipe che non ha la minima conoscenza dei personal

Qualche tentativo era stato fatto, almeno superficialmente, per ricostruire l'atmosfera di Las Vegas '87: da un nucleo centrale di salette per gli incontri, si irradiava tutta una serie di stazioni di lavoro (ma anziché essere aperte e di facile accesso erano dei cubicoli claustrofobici) ciascuna delle quali era progettata per illustrare uno specifico aspetto della versatilità dell'Amiga.

La parola chiave è "uno": in un cubicolo battezzato Desktop Video, alcuni rappresentanti della Aegis davano dimostrazione di un programma costruito con l'aiuto del loro *Videoscape 3D*; nella

Poiché ogni stazione di lavoro metteva in mostra software e hardware che era già sul mercato almeno da un anno, non è sorprendente che la Commodore non sia riuscita ad attirare l'attenzione del pubblico. Ciò che sorprende di più è che la società (o il suo ufficio pubbliche relazioni) sia stata disposta a spendere mi-

gliaia di dollari per un simile non-evento.

Un consiglio per il futuro

Si sarebbe potuta ottenere un'attenzione molto maggiore affiggendo manifesti pubblicitari lungo la strada che porta a Chicago, o spendendo il denaro sprecato nel Comdex per una serie di spot televisivi.

È una cosa che ha grande importanza, per noi che abbiamo investito tempo, denaro e dedizione nell'Amiga. È importante che questo computer sopravviva. E vedere le sue possibilità di essere ben accolto nel mondo professionale rovinate in questo modo proprio da coloro che vi hanno investito più di tutti... be', viene proprio da chiedersi se c'è un futuro per l'Amiga. Una partecipazione così insignificante al Comdex è persino peggio di una mancata partecipazione.

A proposito del futuro: dato che il Comdex si rivolge al mondo professionale, ci aspettavamo di non vedere traccia del C-64 e del C-128, e infatti così è stato. Correva però una voce, che emergeva anche tra le righe degli annunci ufficiali Commodore, secondo cui il C-128 uscirà dalla produzione la prossima estate. Non sarebbe affatto sorprendente, visto che nessun nuovo software gestionale, anzi, nessun nuovo programma specifico per il C-128, appare più da parecchio tempo.

Ma torniamo all'Amiga. È evidente che questo computer, come qualunque altro sistema, non potrà mai prevalere sui PC IBM, che hanno messo ormai solide radici nel mondo professionale. Ma vi sono alcune incrinature nell'armatura dei PC, e la Commodore dovrebbe imparare a puntare su queste debolezze.

Dovrebbe cercare di dimostrare che persino il poco costoso Amiga 500 può lasciarsi alle spalle i più potenti PC per quanto riguarda sonoro e grafica. Dovrebbe puntare sul fatto che l'Amiga 2000 dotato della Bridgeboard può combinare il meglio dell'Amiga e di un PC, e diventare una piccola ed efficiente workstation (e dovrebbe vendere la scheda Bridgeboard a un prezzo più realistico: attualmente è più costosa di un clone del PC/AT).

I "claustrofobici cubicoli" affidati alle software house per le loro dimostrazioni

Dall'altra parte della sala, un'azienda che si chiamava WYSIWYG stava dando dimostrazione di un word processor MS-DOS che veniva venduto a un prezzo di 695 dollari. Se la Commodore avesse invitato al Comdex la Micro-Systems Software (una delle prime software house che si sono dedicate ai suoi computer), avrebbe potuto dimostrare che il word processor *Excellence!* offre prestazioni dello stesso livello (fuorché per quanto riguarda le lingue straniere) per un prezzo che non raggiunge nemmeno la metà di quello MS-DOS.

L'Amiga ha enormi potenzialità, e soltanto la Commodore sembra non accorgersene. Ciò che deve fare è puntare tutto su queste potenzialità lasciando perdere quelle aree (come fogli elettronici e database) dove i PC hanno i loro punti di forza. Poi dare la massima pubblicità a questi elementi di vantaggio. Per usare il linguaggio militare, dovrebbe fare una manovra di attacco sui fianchi.

Conclusioni

Forse un giorno o l'altro la Commodore imparerà. Nel frattempo dovrebbe smetterla di gettare al vento tutti quei soldi per le pubbliche relazioni con le aziende, e investirne una parte per mantenere quella base di utenti che l'ha sostenuta fino a oggi.

Malgrado la magra figura della Commodore al Comdex, ci sono interessanti novità per l'Amiga. Al mio ritorno a casa, infatti, per prima cosa ho aperto la posta e tutti i pacchetti che erano arrivati negli ultimi giorni. Ho trovato tra l'altro *Deluxe Paint III* della Electronic Arts (che ora fornisce anche la possibilità di costruire animazioni), *Photon Paint 2.0* della MicroIllusions, *Rampage* della Activision, *Zork Zero*, *Battletech*, *Journey* e *Shogun* della Infocom, *Twilight's Ransom* della Paragon.

Anche se la Commodore trascura o maltratta la sua creatura, possiamo almeno rallegrarci del fatto che la sopravvivenza dell'Amiga sia affidata in parte anche alle abili e amorevoli mani dei creatori di software. ■

LE SCHEDE BRIDGEBOARD XT A2088 E AT A2286 PER L'AMIGA

Due computer e tre monitor in un solo contenitore: questo è quanto ci viene messo a disposizione dalle schede Janus per spalancare le porte dell'MS-DOS all'Amiga 2000

di Dario Greggio

In questo articolo analizziamo l'ultima e più completa soluzione proposta dalla Commodore per rendere l'Amiga MS-DOS compatibile. Si tratta della scheda di emulazione IBM XT A2088 e della recente IBM AT A2286, due veri e propri computer su scheda, pronti per essere inseriti nell'Amiga 2000.

Riepiloghiamo brevemente i momenti della vicenda Amiga/MS-DOS, che inizia poco dopo l'uscita del primo Amiga 1000. Ben presto, infatti, la Commodore si rese conto che, in un mercato dominato dal sistema operativo MS-DOS, persino un computer molto dotato come l'Amiga non avrebbe potuto permettersi di esporre i suoi utenti a una scelta dolorosa tra la grafica stupenda di una macchina ancora con poco software e le prestazioni per nulla eccezionali ma ben collaudate di un sistema tra i più diffusi. Si prese allora la decisione di offrire la compatibilità con l'MS-DOS come una caratteristica accessoria del nuovo computer. È un po' quello che sta succedendo oggi nel mondo informatico con il sistema operativo Unix, anche se a un altro livello. Comunque, sta di fatto che furono proposte diverse soluzioni.

Dapprima il *Transformer*, un pacchetto software che emulava un PC XT tramite l'hardware e il software di base dell'Amiga. Purtroppo quel prodotto confidava un po' troppo nelle capacità del 68000, e le sue prestazioni erano piuttosto limitate. Poi venne il *Sidecar*, un completo computer IBM compatibile in scala ridotta, da affiancare esternamente all'Amiga 1000.

Da quel momento si cominciò a intravedere quale sarebbe stata la strada che la Commodore avrebbe imboccato per unire i due mondi: la creazione di un compu-

ter a sé, autonomo nell'intero processo di elaborazione, ma in grado di sfruttare i dispositivi di input e di output dell'Amiga. In seguito la tecnologia ha fatto altri passi avanti, e il *Sidecar* ha trovato alloggio all'interno dell'Amiga 2000 con il nome di scheda Janus o A2088.

Questa scheda, già sul mercato da diverso tempo, è un XT-base uscito quando ormai questa macchina era quasi scomparsa, soppiantata dai vari XT "turbo" e dalle loro prestazioni sempre migliori. L'A2088 appariva quindi poco al passo con i tempi ed era abbastanza costosa, due elementi che probabilmente hanno scoraggiato molti potenziali acquirenti. Comunque, dal punto di vista tecnologico ha dimostrato che seguire la via bridgeboard alla ricerca di migliori prestazioni avrebbe dato con ogni probabilità buoni risultati. Da qui è nato l'investimento che ha portato alla recente A2286, un AT che invece si discosta ampiamente dall'AT-base, offrendo prestazioni non eccezionali ma almeno al passo con i tempi.

Dal momento che con questo articolo, oltre che illustrare l'ultimogenita bridgeboard, intendiamo fare il punto sulla situazione in materia di compatibilità IBM/Amiga, torneremo ancora sulla A2088, nonostante sia in catalogo già da parecchio tempo. Questo ci permetterà di completare il panorama delle alternative oggi disponibili, e di esaminare la possibilità di una nuova A2088 turbo, dalle prestazioni allineate con gli attuali XT di cui il mercato sovrabbonda.

Un XT da 512K...

La scheda A2088 è un completo computer IBM XT compatibile dalle dimensioni notevolmente ridotte. Sullo spazio di una scheda PC trovano posto un microprocessore Intel 8088, 16 chip di memoria RAM per un totale di 512K, uno zoccolo per coprocessore matematico 8087, un adatta-

SCHEDE BRIDGEBOARD: PRO E CONTRO

Con le schede bridgeboard l'Amiga può utilizzare i programmi scritti per i personal IBM *senza alcun problema di compatibilità*. Tutte le nostre prove confermano che le schede A2088 e A2286 costituiscono effettivamente un XT e un AT perfettamente compatibili e funzionanti. È apprezzabile soprattutto il fatto che il PC sia subordinato in modo intelligente all'interfaccia grafica dell'Amiga, aggiungendo dunque le sue potenzialità all'ambiente MS-DOS senza intaccare l'indispensabile compatibilità.

Tenendo conto di queste valutazioni, la domanda alla quale cercheremo di rispondere è se sia realmente vantaggioso per un utente dell'Amiga 2000 acquistare una scheda bridgeboard, o se non sia più conveniente acquistare un PC a parte. La nostra stima viene fatta in parte sulla base di una considerazione quasi banale: chi acquista un Amiga 2000 anziché il più economico Amiga 500 lo fa anche per sfruttare i vantaggi che può trarre dalla sua elevata espandibilità.

La prima scelta da fare è quella tra IBM XT e IBM AT. Il primo è un computer piuttosto vecchio come concezione, ma è per esso che sono stati scritti tutti i grandi programmi di successo che oggi affollano il mondo professionale. Sono programmi che difficilmente usano più di mezzo mega di memoria e che fanno in genere affidamento su un hard disk da 20 MB per la memoria di massa, ma nonostante queste limitazioni svolgono il loro lavoro con efficienza. Il problema è che le loro prestazioni in termini di velocità (non di affidabilità, si badi bene) sono spesso mediocri, e ciò si fa sentire sia nel word processing sia nei fogli elettronici, per non parlare della grafica.

L'AT è invece una macchina più versatile e moderna. Grazie al suo microprocessore, l'Intel 80286, può lavorare in multitasking e arrivare perfino a gestire una rete di utenti... anche se in genere viene utilizzato soprattutto per soddisfare le esigenze appena menzionate a una velocità maggiore. Comunque, per dovere di cronaca occorre sottolineare che mentre gli XT possono funzionare soltanto in MS-DOS, gli AT, proprio per le loro più elevate prestazioni, possono funzionare anche con sistemi operativi come l'OS/2 e lo XENIX. Si tenga però presente che un AT costa quasi il doppio di un XT. Questo è vero sia che si parli di un originale IBM, sia che ci si affidi ai cloni fabbricati a Taiwan, sia che si acquisti un PC compatibile della Commodore.

Fatte queste premesse di carattere generale, è doverosa una considerazione che riguarda entrambe le schede bridgeboard: chi possiede già un Amiga 2000 probabilmente non è entusiasta di sprecare altro spazio, e in quest'ottica la scheda bridgeboard è una scelta che presenta diversi vantaggi. Senza parlare dell'interessante sinergia fra le due macchine, sui cui ci siamo ampiamente dilungati nell'articolo.

Per quanto riguarda l'aspetto economico, la **A2088**, la bridgeboard XT, è a nostro giudizio abbastanza costosa, soprattutto perché priva di caratteristiche ormai standard nella stragrande maggioranza degli XT, come per esempio un clock più elevato e 640K di RAM.

Forse potrà essere utile un confronto con il prezzo di listino di una macchina completa. Abbiamo evitato volutamente di scegliere marche sconosciute che non danno alcuna garanzia di affidabilità, ma abbiamo comunque scelto una casa che vanta prezzi piuttosto concorrenziali pur avendo raggiunto ormai una solida reputazione: la Amstrad. Il suo PC2086 SD12MD XT compatibile (con monitor monocromatico, 640K, scheda VGA e mouse, 8 MHz) costa un milione e 499 mila lire contro il prezzo di un milione e 50 mila lire della scheda A2088 (con un disk drive da 5,25", 720K, compatibilità grafica CGA, 4,77 MHz). Il divario aumenta se si prende in considerazione il PC2086 SD12HRCD che a differenza del modello precedente viene venduto con un monitor a colori a un prezzo di 2 milioni e 99 mila lire. Tutti i prezzi che citiamo non comprendono l'Iva.

La valutazione della **A2286**, la bridgeboard AT, è assai diversa. Le sue prestazioni, anche se non superlative, appaiono proporzionate al prezzo e pari a quelle delle altre macchine AT presenti oggi sul mercato. Certo, la A2286 è pur sempre una scheda, ma se già si possiedono il monitor e il cabinet del proprio Amiga c'è da chiedersi se valga la pena di ricomprarli con l'AT.

Per quanto riguarda il prezzo, la A2286 (con un disk drive da 5,25", 1,2 MB, compatibilità grafica CGA, 8 MHz) costa un milione 985 mila lire, mentre un corrispondente modello Amstrad AT compatibile, il PC2286 DD12MD (con monitor monocromatico, scheda VGA, due disk drive da 3,5" e 1,4 MB, mouse, 12 MHz) costa 2 milioni 599 mila lire, mentre la versione con monitor a colori costa 3 milioni 199 mila lire. Ognuno può fare i conti da sé.

La mancanza di un monitor a fosfori verdi, se proprio se ne sente il bisogno, può essere colmata con l'acquisto di una scheda grafica e di un monitor Commodore 1402: il prezzo complessivo non supererebbe quello dei computer AT compatibili Amstrad che abbiamo citato.

tore video che emula una scheda MDA e una CGA, 128K di RAM "dual port" (bifronte, ecco da dove viene il nome Janus) e alcuni chip aggiuntivi dedicati al colloquio tra il bus PC e il bus Amiga. Oltre alla scheda, il kit comprende un disk drive da 5,25'' da installare all'interno dell'Amiga (360K e 720K), un connettore di collegamento fra la scheda e il disk drive che consente di collegare un solo disk drive interno, i manuali, tre dischi da 5,25'' e un disco da 3,5''.

Come indica il nome, la scheda costituisce un ponte (bridge) tra il mondo PC e il mondo Amiga, dotando il primo di due importantissimi "organi di senso": l'input e l'output. La cosa più curiosa è che il ponte viene creato proprio fisicamente, si tratta cioè di una scheda che viene inserita a cavallo tra il bus Amiga e quello PC. Come molti sapranno, infatti, due dei cinque connettori del bus Amiga (non teniamo ovviamente conto del CPU slot) sono allineati con altrettanti connettori del bus PC, proprio per consentire una comunicazione di questo tipo.

La scheda Janus, come dicevamo, non va considerata una semplice periferica, ma un computer vero e proprio. Vi troviamo infatti tutto ciò che è basilare per il funzionamento della CPU 8088 a 4,77 MHz (BIOS Phoenix, RAM, logica di controllo del bus...) e inoltre alcune sezioni destinate a mantenere efficiente l'interazione con l'hardware dell'Amiga. Stiamo parlando essenzialmente dei due chip che interfacciano il bus Amiga con il bus IBM e della RAM dual port.

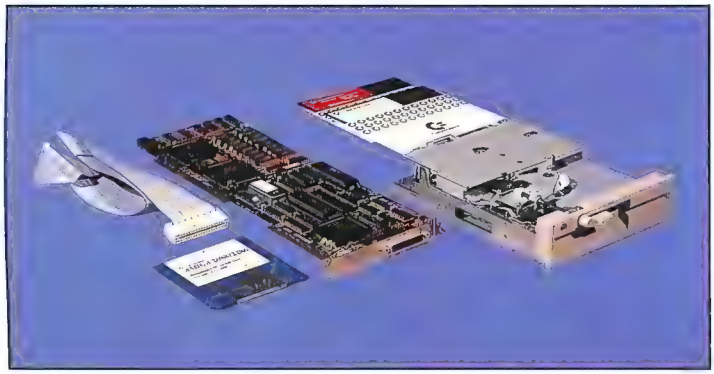
Il compito dell'interfaccia tra i due bus è quello di trasformare gli indirizzi e i dati del bus 68000 in quelli relativi al bus 8088, i quali sono diversi sotto parecchi aspetti, non ultimo quello delle temporizzazioni. Questo permette una comunicazione a livello di singoli dati e indirizzi, mentre per le grosse quantità di dati esiste eventualmente la già citata RAM dual port.

Questa RAM, il cui impiego primario è il parcheggio temporaneo dei dati che andrebbero visualizzati sul monitor, ha la capacità di essere letta e scritta in due modi diversi dalla CPU 8088 e dal 68000. Per fare un esempio, possiamo esaminare la sua gestione nell'ambito delle informazioni video. Quando l'8088 deve mostrare dei dati sullo schermo, accede a particolari locazioni di memoria che sono lo "specchio" dello schermo così come appare sul monitor; il chip video di un normale computer IBM PC è in grado di leggere quest'area di memoria e visualizzare i caratteri o la pagina bit-map corrispondente.

Nel caso della scheda Janus, tuttavia, poiché il monitor dell'XT dev'essere emulato in una finestra dell'Amiga, occorre che a leggere queste informazioni video sia il 68000 e indirettamente il chip Denise che gestisce il video: quindi, tale zona di memoria dev'essere visibile per ambedue gli ambienti. Poiché però il tipo di gestione del video utilizzata nel mondo IBM (schermi grafici "tradizionali", con tutti i byte o i nibble relativi a ciascun pixel posti l'uno di fianco all'altro) differisce notevolmente da quella impiegata nel mondo Amiga (bitplane distinti per ciascuno dei bit relativi a un pixel), si è fatto in modo che fosse l'hardware della scheda a uniformare i due tipi di gestione, liberando il software (del 68000) da un'operazione di decodifica piuttosto laboriosa. Quindi, quando l'Amiga accede alla RAM dual port, ottiene i dati prodotti dall'XT in un formato già adatto alla visualizzazione sugli schermi e sulle finestre di Intuition.

Si tenga comunque presente che tale area di RAM serve anche come buffer per le operazioni di accesso condiviso alle memorie di massa dei due sistemi. Inoltre, la flessibilità della A2088 e del software Amiga di gestione è tale che si può emulare sia uno schermo MDA contenente solo testo monocromatico, sia uno schermo CGA in grado di visualizzare testi con un massimo di 16 colori oppure grafica 320 x 200 con un massimo di quattro colori; i colori sono sempre scelti dall'intera tavolozza dell'Amiga. I due schermi possono essere attivi contemporaneamente (è come possedere due monitor), e per esigenze particolari è sempre possibile, disattivando l'emulazione interna, aggiungere in uno slot PC una scheda grafica e collegarvi un altro monitor interamente dedicato al PC.

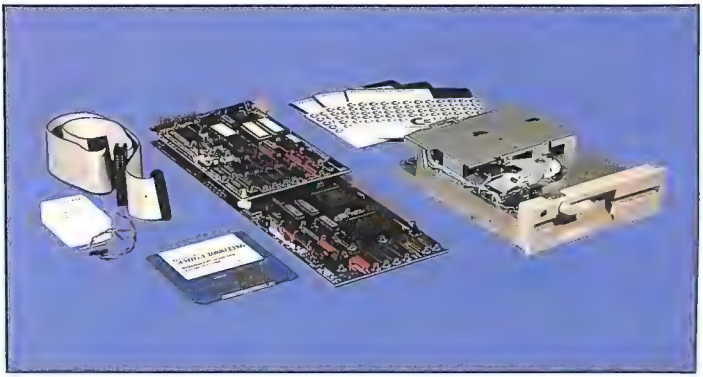
Sulla scheda non vi sono interfacce parallele o seriali: il BIOS che lavora all'interno della A2088 può utilizzare la porta Centronics dell'Amiga come porta parallela.



Gli elementi che costituiscono il kit A2088. Si noti il connettore per il disk drive esterno

mentre per la porta seriale è necessario installare una scheda apposita in uno degli slot PC rimasti liberi. L'unico jumper presente nella A2088 serve per selezionare il tipo di output video attivo al momento dell'accensione del sistema (monocromatico, a colori nei modi 40 x 25 o 80 x 25, o infine nessuna emulazione grafica). Va notato che questa scheda dispone di un connettore, che si affaccia sul retro del computer, adatto al collegamento di un disk drive esterno, eventualmente da 3,5'' (peculiarità questa che viene perduta nella A2286).

Infine, per quanto riguarda il suono la scheda A2088 non è dotata di altoparlante, ma prevede un connettore (J2) al quale è possibile collegarne uno per proprio conto. Nella nostra prova abbiamo collegato un altoparlantino da 8 Ω e il sonoro ha subito funzionato. Sarebbe stata preferibile una soluzione di serie integrata sulla scheda, ma probabilmente avrebbe incontrato problemi di spazio, mentre in questo modo l'altoparlante separato può essere installato nell'Amiga dove si preferisce.



Gli elementi che costituiscono il kit della A2286. Si noti la scheda a due basette

L'utility SETUP per variare la configurazione di default

...e un AT a 8 MHz

La scheda A2286 è più affollata di una spiaggia in pieno agosto, tanto che per ricavare un po' di spazio aggiuntivo è stata realizzata una doppia basetta. Come si vede dalla foto di pagina 73, la scheda principale contiene i due chip incaricati del colloquio con il bus dell'Amiga, la CPU 80286, lo zoccolo per l'eventuale coprocessore matematico 80287, mentre la scheda minore (definita "sandwich" nel manualetto allegato) contiene 1 MB di RAM e alcuni altri chip, tra cui la memoria dual port usata come buffer durante l'interscambio dei dati tra i due mondi. Il kit comprende oltre alla scheda un disk drive da 5,25'', 360K, 720K e 1,2 MB, un connettore per collegare fino a due disk drive interni e gli stessi manuali e dischi inclusi nel kit della A2088.

Ancor più della A2088, questa scheda va considerata un secondo computer sofisticato quasi quanto l'Amiga. Se le caratteristiche della A2088 erano quelle di un PC XT-base e nulla più (cosa che provocò molte polemiche al momento della presentazione), la A2286 rappresenta invece un deciso passo avanti rispetto all'IBM AT-base. Il suo microprocessore 80286 funziona a 8 MHz (un terzo più veloce dell'AT-base) e la RAM disponibile offre i 640K canonici più altri 384K di memoria estesa; è inoltre presente un orologio con batteria tampone (batteria che mantiene anche la configurazione interna di default). La memoria estesa è quella indirizzabile oltre il primo megabyte, e si tratta di una memoria non visibile direttamente dall'MS-DOS. A questo proposito si veda la prova del PC40-III su questo stesso numero della rivista, nella quale vengono esaminati alcuni problemi relativi alla gestione della memoria nel mondo MS-DOS.

Anche la scheda A2286 incorpora un controller che può gestire un massimo di due disk drive (esclusivamente interni) da 360K, 720K e 1,2 MB nel formato da 5,25'', oppure da 360K, 720K e 1,44 MB nel formato da 3,5''. Un altro parallelismo si nota nel modo in cui l'output video viene emulato sul lato Amiga della macchina: come nel caso precedente, l'adattatore interagisce con un'area di RAM dual port, che viene scritta e letta da ciascuno dei due computer in maniera diversa.

Già al momento dell'installazione della scheda AT, si nota purtroppo un particolare negativo, cioè ci si accorge che essa in realtà occupa lo spazio di due slot PC/Amiga e non di uno solo a causa dello spessore della schedina aggiuntiva. La cosa non è troppo grave, in quanto nell'Amiga non si può montare più di un emulatore alla volta e quindi sarebbe inutile disporre di un

secondo ponte. Ma è anche vero che gli slot coperti avrebbero potuto essere utilizzati separatamente. In ogni caso, conviene inserirla nello slot PC/Amiga più vicino all'alimentatore, dal momento che in questa posizione finisce per compromettere solo uno slot Amiga.

Non vi sono comunque modifiche hardware da apportare alla macchina, e dopo aver esaminato la funzione dei jumper presenti sulla scheda A2286 possiamo passare al collaudo. Esiste un jumper, J5, per attivare o disattivare la memoria estesa; J4 serve per abilitare o meno il controller interno del disk drive; J14 serve per fare in modo che all'accensione risulti attiva l'emulazione in monocromatico o quella a colori. Di alcuni jumper, purtroppo, non viene data descrizione; in particolare, non viene detto che J2, come sulla scheda A2088, è l'uscita audio alla quale possiamo collegare un altoparlante. Anche per questa scheda abbiamo tentato il collegamento con un altoparlantino da 8 Ω e tutto ha funzionato perfettamente.

Segnaliamo infine che il BIOS della A2286 incorpora una routine di setup (richiamabile con la pressione dei tasti Control-Alt-Esc) del tutto identica a quella presente nel PC40-III: questa routine consente d'impostare diversi parametri di default del sistema.

La doppia personalità

Accendiamo l'Amiga e andiamo a vedere che cosa succede. Dal momento che il funzionamento delle due schede, in ambito MS-DOS, è assolutamente identico, la nostra prova le considera congiuntamente. L'unica cosa che cambia sono le prestazioni. Precisiamo inoltre che l'Amiga utilizzato per la prova delle due bridgeboard conteneva la scheda acceleratrice A2620 (di cui si è parlato nello scorso numero di *Commodore Gazette*) e disponeva, dal lato PC, di un hard disk da 20 MB (suddiviso in parti uguali tra i due sistemi operativi), oltre che di una scheda grafica AGA funzionante in emulazione MDA monocromatica collegata a un monitor a fosfori verdi. Con tutte queste periferiche in dotazione, è stato possibile valutare le prestazioni dell'emulazione MS-DOS sull'Amiga in un gran numero di configurazioni diverse. In particolare, per utilizzare il controller PC e l'hard disk (PC5060+A2092) non abbiamo dovuto toccare alcun jumper sulle bridgeboard.

La prima volta che si esegue il boot dell'Amiga occorre utilizzare il disco PCInstall, fornito con le schede, il quale contiene il software necessario per la comunicazione tra i due mondi (si noti che per entrambe le schede viene fornito lo stesso disco d'installazione, a dimostrare che dal punto di vista software, ma soprattutto agli occhi dell'Amiga, non ci sono differenze fra la A2088 e la A2286).

La finestra MS-DOS aperta sullo schermo del Workbench

L'installazione consiste semplicemente nella creazione di un disco *Workbench* che contiene anche tutti i file necessari per interagire con la bridgeboard installata; nelle successive sessioni di lavoro ci si servirà di questo disco. Ovviamente, l'installazione può anche avvenire su un hard disk.

Per eseguire l'installazione, si deve ricorrere a una delle quattro procedure presenti nella directory radice del disco PCInstall: BridgeInstall (A2000 con 1 MB di RAM), BridgeInstall512 (A2000 con 512K), MiniInstall (A2000, copia soltanto le librerie, le fonti e i dispositivi necessari al funzionamento della bridgeboard, mentre non copia tutti i programmi di gestione della scheda), SidecarInstall (A1000 più Sidecar). MiniInstall è particolarmente comoda se si possiedono due disk drive nel proprio Amiga.

Attivando una di queste procedure, appare una finestra che elenca i file da copiare per default: l'utente può accettare la cernita di default, oppure alterarla per togliere o aggiungere altri file. Si può poi, dopo aver confermato la scelta, indicare come destinazione un floppy o un hard disk, dopo di che la procedura controlla che vi sia sul supporto di destinazione spazio libero a sufficienza (avvisando eventualmente che occorre cancellare qualcosa per fare spazio: anche in questo caso viene fornita una scelta di default di file da cancellare).

Quando si è creato spazio sufficiente viene automaticamente eseguita la copia dei file, e alla fine disponiamo di un disco o di un hard disk adatto al boot del sistema MS-DOS. Si noti che la procedura di copia fa uso del RAM disk, e quindi in un computer con 512K la copia deve avvenire in due fasi. Occorre precisare che l'intera procedura ha lo scopo di permettere l'installazione del software per la scheda su un disco sistema di cui già ci serviamo ampiamente. In effetti, per usare la scheda si potrebbe anche ricorrere a una semplice copia del disco PCInstall, con lo svantaggio però che molto spazio rimarrebbe occupato da file non più utili.

Il software di gestione della bridgeboard consiste in un dispositivo (janus.device) e in una libreria (janus.library), oltre a due

utility di attivazione dei monitor simulati e a diversi tool la cui funzione verrà esaminata in seguito. La comunicazione tra i due mondi avviene attraverso interrupt hardware e scambio di dati in DMA. In particolare, se nella startup-sequence è presente il comando BINDDRIVERS, questo "interroga" tutte le schede che rileva sul bus d'espansione per farsi dire i nomi delle relative librerie di gestione; poi carica in memoria e rende attive tutte le librerie di cui ottiene il nome. Nel caso della scheda Janus, che ovviamente risulta presente sul bus dell'Amiga, BINDDRIVERS ottiene il nome "janus.library", la libreria di gestione della bridgeboard. Questa libreria viene quindi caricata dal disco sistema e attivata. Fino a questo momento, la scheda PC non ha ancora dato segni di vita.

A questo punto, la libreria si preoccupa d'inizializzare la scheda: fra le varie operazioni che compie, carica dal disco sistema un file di nome PC.Boot e lo trasferisce nella RAM della scheda. Solo ora il PC prende vita e inizia la propria procedura d'inizializzazione.

Prima ancora che il boot dal lato Amiga sia terminato, il BIOS sulla scheda bridgeboard (attivato dalla libreria Janus) ha già terminato la sua fase di startup e se abbiamo inserito il disco sistema MS-DOS nel disk drive da 5,25'' è stato effettuato anche il boot. Di queste operazioni, però, non veniamo avvisati in alcun modo.

Il software MS-DOS fornito con le due schede consiste in tre dischi, i primi due contenenti il software sistema tipico di ogni PC compatibile (MS-DOS e GW-BASIC), e il terzo contenente una directory JANUS nella quale troviamo una serie di utility specificamente destinate alla gestione della scheda bridgeboard. Vengono inoltre forniti i manuali del sistema operativo e dell'interprete Basic, oltre alla guida per l'utente (uguale per entrambe le schede) che pur essendo in inglese è molto facile da consultare, grazie alle fotografie che illustrano le varie fasi dell'installazione hardware e software.

La finestra PCWindow

Anche se l'ambiente MS-DOS è ora pronto a ricevere comandi, il problema è che non ha dispositivi di input o di output e quindi è nostro compito fornirglieli. Fino a quando non desideriamo entrare nel mondo MS-DOS, possiamo tranquillamente dimenticarci che all'interno dell'Amiga c'è un completo PC XT o AT in funzione, pronto per essere impiegato in qualsiasi momento.

La finestra PCWindow e una seconda finestra MS-DOS

Quando desideriamo accedere al PC, è sufficiente mandare in esecuzione (dal *Workbench* o dal CLI) una delle due utility PC Color o PC Mono: la prima attiva lo schermo a colori, la seconda lo schermo monocromatico. In entrambi i casi si apre una finestra di Intuition, di nome PCWindow, dedicata all'MS-DOS: a questo punto (e finché non chiudiamo la finestra o non la deselezioniamo) le nostre interazioni con la tastiera giungono alla scheda bridgeboard, e l'output video da questa generato appare nella finestra di Intuition. In pratica, ora abbiamo un nuovo task di comunicazione con il PC.

È bene esaminare attentamente le caratteristiche della finestra attivata per il nuovo sistema operativo. Quella relativa allo schermo monocromatico può avere due o quattro colori, come al solito scelti fra quelli dell'Amiga, che andranno a sostituire i due colori di schermo e di fondo del modo monocromatico dell'MS-DOS, oppure i due colori più gli attributi evidenziato e lampeggiante; naturalmente questi attributi perdono i loro significati originari e vengono rimpiazzati da due nuovi colori.

La finestra relativa allo schermo a colori invece può andare da due a sedici colori, tanti quanti sono quelli disponibili con una scheda CGA in modo testo. In entrambi i casi, cambiando il numero di colori disponibili nella finestra MS-DOS mutano le caratteristiche della finestra stessa. Quando i colori sono esattamente quattro, la finestra può essere aperta sullo schermo del *Workbench*, mentre in tutti gli altri casi necessita di uno schermo personale. Questo avviene indipendentemente per ognuno dei due pseudo-monitor attivabili, che possono anche essere aperti contemporaneamente.

Che sia quella monocromatica o quella a colori, la finestra che si apre agli occhi dell'utente è anche dotata di una barra menu, tramite la quale l'utente può svolgere operazioni di gestione dell'input e dell'output che intervengono solo sul lato Amiga, e non influenzano il comportamento della scheda bridgeboard. Questa sovrapposizione dell'interfaccia grafica di Intuition con lo schermo del PC non deve destare sospetti. I due ambienti rimangono sempre ben distinti, e Intuition serve solo per rendere più gradevole l'interazione con il PC.

Le voci del menu consentono d'impostare il numero di colori desiderato, di sceglierli mediante

una palette Amiga (possibilità del tutto inedita per un utente MS-DOS), di spostare la finestra da uno schermo all'altro, di congelarne l'output (attenzione, non l'elaborazione MS-DOS sottostante), di portarne le dimensioni in un sol colpo a tutto schermo e viceversa, di eliminare i bordi laterali e il gadget di dimensionamento per visualizzare correttamente le 80 colonne dei modi grafici MS-DOS, altrimenti mutilate (queste due funzioni si possono ottenere anche con un doppio click del mouse all'interno della finestra).

Fra le altre opzioni disponibili, una consente di alterare la priorità del task Amiga adibito all'aggiornamento dello schermo MS-DOS, al fine di avvantaggiarlo o penalizzarlo nei confronti degli altri task eventualmente in esecuzione sul lato Amiga. Teniamo a precisare che la velocità della scheda bridgeboard non viene compromessa dal rallentamento nell'esecuzione di quel task: ne soffre infatti soltanto la velocità di aggiornamento del video, la quale, in caso di animazioni o scroll, presenta alcune anomalie.

In realtà a noi è capitato di constatare un rallentamento dell'elaborazione in MS-DOS quando abbiamo attivato contemporaneamente l'adattatore video esterno di cui disponevamo lasciando in funzione anche quello emulato per utilizzare lo schermo dell'Amiga: ci è sembrato che le due schede rallentassero esclusivamente a causa della doppia gestione, in quanto in tutti gli altri casi il fenomeno non si verifica.

Infine, è anche presente un help che descrive succintamente le peculiarità di ogni opzione, e la possibilità di salvare la configurazione impostata (colori, velocità di risposta della tastiera, priorità del task di aggiornamento dello schermo...), così da renderla di default.

Il cut & paste, un'alternativa ai file batch

Sempre tramite il menu, è anche possibile eseguire funzioni di cut & paste tra i due mondi facendo uso del dispositivo Clipbord dell'Amiga, naturalmente nel modo più immediato e cioè tramite mouse: un testo "ritagliato" dallo schermo PC finisce nel Clipboard, e un testo presente nel Clipboard può essere inviato al PC proprio come se venisse digitato da tastiera. Questa possibilità ci consente, per esempio, di creare un'alternativa ai file batch dell'MS-DOS: in breve, i file batch consentono di automatizzare sequenze di comandi al livello della linea comandi, mentre impiegando l'opzione paste possiamo fare in modo che in qualsiasi posizione si trovi il cursore, anche all'interno di applicazioni, appaia il testo precedentemente salvato con l'opzione cut. Abbiamo per esempio mandato in esecuzione sul PC il word processor MS-DOS *XYWrite* e, dopo essere entrati nella linea comandi, abbiamo eseguito il paste di un testo precedentemente salvato con il tool Notepad del *Workbench*, nel quale avevamo inserito una serie di comandi riconosciuti dal word processor. Tutto ha funzionato perfettamente: i comandi sono stati eseguiti come se li avessimo semplicemente digitati. Peccato soltanto che le operazioni di cut & paste non consentano di trattare con file individuati da un nome.

L'ambiente Windows necessita di uno schermo proprio

Concludiamo questa parte sull'interazione con il mondo MS-DOS attraverso Intuition giudicando positivamente le operazioni che si possono eseguire tramite la finestra PCWindow senza intaccare il comportamento del PC. Alcune sono finezze, ma altre aggiungono possibilità che con un semplice PC, XT o AT, non sono nemmeno pensabili. Quelle che giudichiamo più interessanti sono due. Anzitutto, la possibilità di aprire più finestre PCWindow sullo stesso schermo e congelarvi le informazioni video. Questo corrisponderebbe, se avessimo solo un PC, a un'ipotetica configurazione con più schermi che mostrano diversi gruppi d'informazioni: una situazione di per sé impossibile che con l'Amiga diventa realtà. Per esempio, quando i risultati di un lungo calcolo eseguito con *Lotus 123* sono disponibili sullo schermo virtuale, possiamo congelare quella finestra e collocarla sullo schermo dell'Amiga dove più ci aggrada. Poi apriamo un'altra finestra all'interno della quale continuiamo a lavorare con il PC, magari mandando in esecuzione un word processor per redigere una relazione sui dati ottenuti, dati che sono sempre visibili nella finestra congelata.

La seconda caratteristica che ci sembra interessante riguarda la palette di colori dalla quale possiamo scegliere, con appositi slider RGB, i colori del nostro schermo virtuale MS-DOS. Ovviamente, il ventaglio di colori possibili è quello consentito dall'Amiga, che è senza dubbio elevatissimo in confronto a quello disponibile con i PC-base.

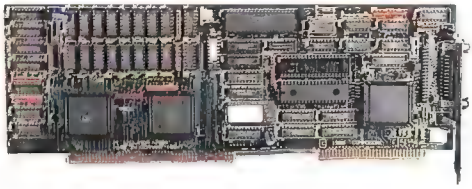
Disk drive e hard disk virtuali

La semplice constatazione che un computer MS-DOS su scheda

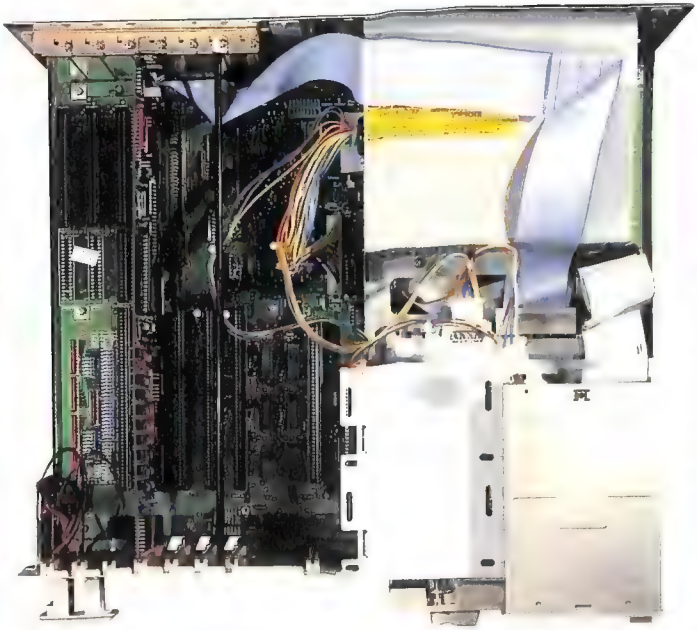
può lavorare all'interno dell'Amiga non è particolarmente interessante. Ciò che invece colpisce è la possibilità per i due mondi di coesistere al punto di potersi scambiare i file, le memorie di massa e i programmi. L'esempio più appariscente e immediato è la visualizzazione da parte dell'Amiga di uno schermo MS-DOS all'interno di una finestra di Intuition, ma esistono molti altri modi di comunicare.

Iniziamo dai drive virtuali, con la premessa indispensabile che per il loro funzionamento è necessario che sull'Amiga sia in esecuzione in background l'utility PCDisk (si può mandare in esecuzione durante la startup-sequence o in qualsiasi altro momento).

Se non si dispone di una scheda controller e di un hard disk da impiegare nel mondo MS-DOS, si può ricorrere a uno stratagemma che permette all'MS-DOS di vedere un hard disk che in realtà è simulato nell'ambiente Amiga. Vediamo com'è possibile. Il trucco è creare su uno dei dispositivi di massa dell'Amiga un file che dal lato MS-DOS viene visto come un hard disk. Per crearlo, l'utente deve mandare in esecuzione dal CLI l'utility MakeAB, presente sul disco sistema della bridgeboard. Questa utility rivolge all'utente tutta una serie di domande sulle caratteristiche che l'hard disk virtuale deve possedere, come il numero di testine, di cilindri e così via. Ottenute queste informazioni, provvede a creare un file (anche in FFS) le cui dimensioni corrispondono all'intero hard disk richiesto e dotato del nome indicato dall'utente. In questo file finiranno tutti i file che dal mondo MS-DOS vengono copiati nell'hard disk virtuale. Infine, MakeAB crea anche un file ASCII Aboot.CTRL nella directory PC/system del disco sistema, nel quale inserisce il nome che l'utente ha indicato per l'hard disk virtuale creato da MakeAB. Di volta in volta si possono dunque avere più hard disk virtuali, cambiando semplicemente il contenuto di questo file.



La scheda A2088: si notino le RAM e lo spazio per l'8087



La scheda XT A2088 installata in uno slot bridgeboard dell'Amiga 2000

Ora l'hard disk è pronto: basta entrare in MS-DOS e impartire i comandi FDISK e FORMAT per utilizzare l'hard disk virtuale, che, copiandovi i file di sistema, può anche essere usato come disco per il boot dell'ambiente MS-DOS. L'emulazione è totalmente trasparente per il BIOS della macchina MS-DOS: grazie alle variazioni tecniche che MakeAB consente di operare sulla struttura dell'hard disk virtuale, si possono creare memorie di massa riconosciute dal BIOS o anche non esistenti nella realtà. Questo hard disk virtuale serve per offrire ai possessori di controller e hard disk Amiga (come la combinazione A2090 + A2092) l'opportunità di utilizzarli anche per i file MS-DOS. Non si giunge a una vera e propria partizione MS-DOS, ma il trucco impiegato è del tutto invisibile dal lato MS-DOS. Ci sembra invece molto meno utile utilizzare MakeAB per aprire un hard disk MS-DOS su un disco da 3,5" inserito in uno dei disk drive dell'Amiga, dal momento che la velocità d'accesso risultante è a dir poco ridicola rispetto a quella di un vero hard disk.

Un'altra strada da seguire per avere a disposizione drive virtuali dal mondo MS-DOS è l'uso di JLINK. Il comando JLINK e il dispositivo relativo (JDISK.SYS) risiedono nella directory JANUS del terzo disco sistema MS-DOS, e

consentono di creare fino a quattro dischi virtuali (che vanno ad aggiungersi ai dispositivi di massa già presenti) su uno qualsiasi dei dispositivi di massa dell'Amiga.

La differenza con il metodo precedentemente descritto è che da questi dischi virtuali non si può effettuare il boot dell'MS-DOS; inoltre sono vincolati al sistema operativo, mentre il disco creato con MakeAB può funzionare con altri sistemi operativi. Per crearli, bisogna anzitutto inserire nel file CONFIG.SYS dell'MS-DOS (il file che indica quali driver di sistema sono attivi) la riga DEVICE=JDISK.SYS, il che avviserà l'MS-DOS (al prossimo boot) della presenza di dischi virtuali dal lato Amiga. A questo punto occorre creare i file Amiga corrispondenti alle memorie di massa virtuali: questo si ottiene con il comando JLINK, al quale vanno specificati il path completo del file da creare (dal lato Amiga), le sue dimensioni in Kbyte e il disk drive virtuale MS-DOS a cui dev'essere associato. Analogamente al caso precedente, i file generati dal lato Amiga non sono in formato ASCII, ma possono comunque essere copiati e rinominati dopo essere stati scollegati dal lato MS-DOS.

Dopo che questi drive virtuali sono stati creati e utilizzati, si possono rendere a sola lettura oppure si possono eliminare, agendo dal mondo MS-DOS, mediante una nuova esecuzione del comando JLINK con le opzioni "/r" o "/u". Si può anche fare in modo che eventuali messaggi d'errore generati quando qualcosa non funziona nei dischi virtuali siano soppressi. Naturalmente, i dispositivi Amiga su cui si possono aprire questi dispositivi virtuali sono soltanto quelli in grado di memorizzare dati in lettura e scrittura, anche se il sistema non ci impedisce di usare per esempio la porta seriale (il che non ha senso).

Abbiamo constatato che un eventuale reset del PC mentre sono attivi dischi virtuali di tipo JLINK li rende illeggibili dal lato MS-DOS e inaccessibili dal lato Amiga, cioè irrecuperabili: occorre dunque fare una certa attenzione nell'usare questi dischi, e soprattutto chiuderli prima di terminare una sessione di lavoro o procedere con un reset.

Hard disk reali

Esiste poi la possibilità di utilizzare e partizionare anche dal lato Amiga un hard disk installato sul lato PC: qui andiamo a toccare un discorso abbastanza delicato. Infatti, non volendo acquistare due hard disk diversi per i due computer, potrebbe essere vantaggioso acquistare un hard disk PC che costa meno e si può "vedere" (e partizionare) da entrambi gli ambienti. D'altro canto, da un tale hard disk non si può effettuare il boot dell'Amiga.

Acquistando invece un controller Amiga A2090 o A2090A (la versione autoboot) sorgerebbe un altro problema: dall'ambiente MS-DOS non potremmo accedere all'hard disk direttamente ma solo attraverso uno dei due gestori di disk drive virtuali sopra esaminati (preferibilmente il primo, visto che il secondo non permette il partizionamento né la formattazione e non è del tutto trasparente). Questo non è molto grave visto il tipo di emulazione consentito da MakeAB, però potrebbe lasciare delle perplessità... per esempio il file corrispondente all'hard disk può essere facilmente cancellato per errore, a meno che non si trovi in una partizione dedicata dell'hard disk Amiga, come consiglia il manuale.

Tornando comunque al nostro hard disk PC, osserviamo che può essere partizionato dall'MS-DOS con il comando FDISK, e dall'AmigaDOS con il comando ADISK: i due comandi hanno caratteristiche simili e permettono di creare, cancellare o esaminare una partizione dell'hard disk. Dopo questa operazione, si possono formattare le partizioni ottenute dai rispettivi ambienti e a questo punto tutti i volumi logici saranno visibili. Ricordiamo solo brevemente che in MS-DOS non vengono riconosciuti volumi logici di oltre 32 MB e che l'hard disk potrà essere usato dal lato Amiga anche con il FastFileSystem.

Lo scambio di file e il mouse

Andiamo ora a esaminare le due utility d'interscambio dei file, AREAD.EXE e AWRITE.EXE. Anche per utilizzare questi due programmi, presenti sul disco sistema MS-DOS, dev'essere in funzione il tool PCDisk dal lato Amiga. Con AREAD si può trasferire un file AmigaDOS a uno qualsiasi dei dispositivi MS-DOS (disk drive, schermo, RAM disk...), mentre con AWRITE si svolge l'operazione contraria verso qualunque dispositivo dell'Amiga (file su disk drive, CON:, SER:, PAR:...): la sintassi di entrambi i comandi prevede la descrizione dei due pathname (Amiga e MS-DOS) e l'inserimento di alcuni switch. Tramite queste opzioni possiamo decidere se ci interessa che i caratteri non ASCII siano convertiti negli equivalenti Amiga o PC a seconda dei casi, e se desideriamo convertire le sequenza CR-LF in LF singoli e viceversa: tutto questo serve di solito quando si trasferiscono file di testo, e i manuali spiegano esattamente quali sono le conversioni che possono essere effettuate.

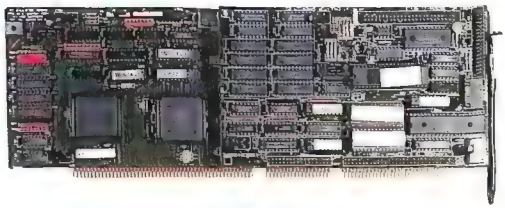
Si noti che da Amiga a PC si possono trasferire soltanto i singoli file, mentre da PC ad Amiga si può anche copiare un insieme di file. Poiché qualsiasi dispositivo di ciascuno dei due ambienti può essere sorgente o destinatario, si può risolvere artificialmente il problema della mancanza di un'interfaccia seriale dal lato MS-DOS, usando per esempio AWRITE con il dispositivo SER: dell'Amiga. L'unica critica che ci sentiremmo di muovere a questi due tool è il fatto che sono dei programmi di trasferimento file e non sono in grado di funzionare come filtri: lavorano cioè sui file

specificati come argomenti anziché "filtrarli" tra un dispositivo e un altro, il che sarebbe consentito sia dall'MS-DOS sia dall'AmigaDOS. Se fossero stati realizzati in questa seconda maniera (precisiamo ancora che la differenza è solo formale e riguarda la praticità d'uso) si sarebbe creato un vero e proprio ponte software.

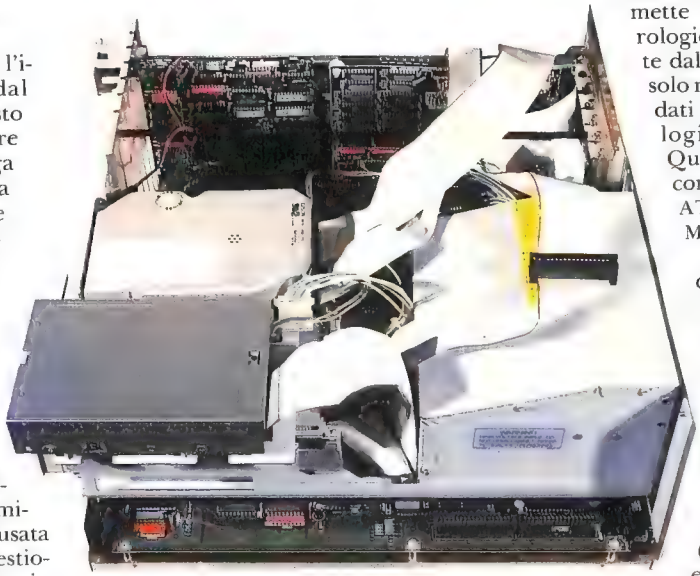
Parliamo ora dell'emulazione del mouse. Il mouse Amiga può essere usato dal lato PC come se fosse un mouse Microsoft o compatibile mandando in esecuzione il comando AMOUSE. Dal lato Amiga bisogna mandare in esecuzione da una finestra CLI il comando:

```
RUN > NIL: PC/AMouse
```

oppure selezionare l'icona AMouse dal *Workbench*. A questo punto si può usare un mouse Amiga nella porta destra per comunicare con il mondo MS-DOS, mentre quello collegato alla porta di sinistra sarà sempre utilizzabile con Intuition. Se, come è probabile, si desidera usare un solo mouse, la combinazione di tasti Amiga-P potrà essere usata per invertire la gestione delle due porte: in questo caso la barra titolo di tutte le finestre PC evidenzia a quale porta è collegato il mouse. Esiste anche un driver, AMOUSE.DRV, che va usato esclusivamente insieme all'applicazione *Windows* della Microsoft: la seconda versione di questo driver, chiamata AMOUSEX.DRV, non ci riguarda in quanto può essere utilizzata soltanto dai possessori del Sidecar.



La A2286: in alto a destra il connettore per i disk drive interni



La scheda AT A2286 installata nell'Amiga insieme a un hard disk

Altre utility

Vediamo ora anche le altre utility presenti nel disco sistema PCInstall. Abbiamo già citato PCDisk, che serve per tutte le procedure di colloquio a livello di file. Esaminiamo ora PCHard, il cui scopo è fornire un reset quasi-hardware alla scheda bridgeboard, nel caso in cui un programma MS-DOS non risponda più alla tastiera. Può naturalmente succedere che anche PCHard non abbia effetto, e in tal caso occorre proprio reinizializzare l'Amiga. A questo proposito, è quasi scontato che se s'impartisce da tastiera il reset dei PC (Alt-Ctrl-Del) viene resettata solo la bridgeboard, e non l'Amiga.

Il tool LPT1 va usato per attivare la porta parallela dell'Amiga come emulazione della porta parallela LPT1: dell'MS-DOS. Se la porta è già utilizzata dall'Amiga l'operazione non viene svolta, altrimenti si apre una finestra con la sola barra titolo che ci indica se la porta è in uso da parte dell'MS-DOS (cosa che esclude l'uso da parte dell'AmigaDOS, che andrebbe in Guru Meditation). Se invece si tenta di stampare qualcosa dall'MS-DOS mentre il tool LPT1 non è attivo, il sistema avverte che la stampante non risulta collegata. Nella directory Services troviamo poi TimeServ, che lasciato residente permette d'impostare l'orologio interno presente dal lato PC (di serie solo nella A2286) con i dati forniti dall'orologio dell'Amiga. Questo si ottiene con le due utility ATIME e ASETTIME dell'MS-DOS.

Infine parliamo di PCPrefs, l'utility che forse più profondamente va a modificare il sistema PC/Amiga. Per suo tramite si può scegliere se disattivare o meno i due monitor simulati (disattivandoli entrambi, in pratica si disabilita il controller video della scheda bridgeboard). Si può scegliere inoltre se la porta seriale dal lato PC può essere usata (questo gadget in realtà non è selezionabile, e nel manuale la sua presenza non viene motivata in nessun modo), e infine a quale indirizzo l'adattatore video trova la memoria di schermo all'interno della RAM dual port. La possibilità di disatti-

vare entrambi gli schermi (monocromatico e a colori) è utile quando si fa uso di una scheda grafica esterna e di un monitor esterno e non si ha bisogno dell'output sul monitor dell'Amiga. In un caso come questo, occorre sempre aprire una finestra PC sullo schermo Amiga, che però funge solo da canale di input per la scheda bridgeboard: l'output della scheda non viene dirottato sulla finestra, che pertanto può essere ridotta alle minime dimensioni per non occupare inutilmente lo schermo dell'Amiga. I vantaggi di questa configurazione con due monitor sono principalmente due.

Tramite la scheda grafica e il relativo monitor possiamo ottenere modi grafici che sugli schermi virtuali dell'Amiga non sono emulati; per esempio, l'ambita grafica VGA non è prevista dal controller video delle due schede bridgeboard, e quindi è ottenibile solo impiegando una scheda grafica e un adeguato monitor aggiuntivo. Inoltre, una scheda grafica con monitor consente anche di vedere l'output video PC senza che sia filtrato dall'Amiga (in particolare, da Intuition), e quindi inevitabilmente rallentato.

Per quanto riguarda invece la possibilità di alterare l'indirizzo della memoria video all'interno del buffer, ci sembra di scarsa utilità, in quanto è troppo strettamente legato all'hardware delle due schede e i giusti valori sono già impostati dalla casa costruttrice: ciò rende inutile e dannoso (leggi blocco del sistema) un intervento errato da parte dell'utente. PCPrefs, una volta decisa la configurazione, permette che venga semplicemente usata oppure salvata su disco come configurazione di default.

Uno sguardo al futuro

Parliamo ora brevemente delle possibilità offerte dall'interazione delle risorse dei due ambienti. Per il momento non ci risulta che esistano applicazioni che facciano un uso condiviso delle due CPU e delle risorse connesse, anche se quest'idea appare molto stimolante. Avremmo in pratica a disposizione un transputer con due CPU abbastanza potenti (nel caso della scheda AT, la 80286 e la 68000) e quindi i tempi di elaborazione verrebbero alquanto ridotti. Il manuale, però, per quanto ben fatto, non scende certo nei dettagli della comunicazione tra i due bus, e noi possiamo solo auspicare che in futuro vengano diffusi i particolari del "miracolo" tecnologico che rende possibile la simbiosi tra i due mondi, affinché nasca qualche applicazione di tipo bridge. Infine, la possibilità di mantenere in funzione sullo schermo dell'Amiga due schermi virtuali dalle differenti caratteristiche grafiche potrebbe trovare il favore di applicazioni predisposte per offrire all'utente parecchie informazioni video contemporaneamente.

Il giudizio tecnico

Il più bel complimento che si può fare alle schede bridgeboard è dire che funzionano esattamente come i computer che intendono riprodurre. Niente più attese come con il *Transformer*: tutto funziona perfettamente e quel che c'è di bello è che l'Amiga resta libero di svolgere altri compiti, pur visualizzando in un angolo il contenuto dello schermo MS-DOS.

I benchmark eseguiti sono gli stessi utilizzati per misurare le prestazioni del PC40-III, che trovate a pagina 53 di questo numero di *Commodore Gazette*, insieme con il relativo listato. I risultati riguardanti la pura velocità di elaborazione hanno dimostrato che le schede sono realizzate praticamente alla stessa maniera e con le stesse caratteristiche tecniche dei computer PC compatibili veri (il PC10-II per la A2088 e il PC40-III per la A2286): a parità di frequenza di clock non si riscontra la minima differenza nei risultati.

L'uso di alcuni programmi di test ha segnalato che il BIOS presente su entrambe le schede bridgeboard riscontra la presenza di una scheda Game Card (per l'uso del joystick), di un'interfac-

XT A2088

Specifiche:
Scheda bridgeboard da installare a cavallo di uno slot Amiga e uno IBM PC, rende possibile l'esecuzione di programmi scritti per la CPU 8086/8088 sull'Amiga.
Coprocessore matematico 8087 opzionale
Microprocessore:
Intel 8088 funzionante a 4,77 MHz
Memoria RAM:
512K su scheda, espandibile tramite schede aggiuntive sul bus PC
BIOS:
BIOS di produzione Phoenix, compatibile con il BIOS IBM XT
Grafica:
Dall'Amiga vengono emulati i modi grafici tipici delle schede MDA e CGA per IBM PC
Memorie di massa:
Un disk drive da 5,25", che lavora con dischi formattati a 360K o 720K
Controller:
In grado di gestire un disk drive interno e uno esterno, da 3,5" o 5,25", con dischi di capacità compresa tra 360K e 720K
Espandibilità:
Possibilità di aggiungere un secondo disk drive esterno.
Sul bus PC dell'Amiga restano liberi tre slot
Dispositivi:
Controller per due disk drive.
Porta parallela Centronics emulata sulla porta Amiga.
Interfaccia per mouse Microsoft o compatibili emulata tramite una delle due porte mouse dell'Amiga
Tastiera:
Tastiera IBM XT emulata con la tastiera Amiga
Dimensioni:
Formato standard delle schede per Amiga 2000
Prezzo al pubblico:
L. 1.050.000 + Iva
Distribuzione:
Commodore Italiana spa
Viale Fulvio Testi, 280
20126 Milano
(Tel. 02/661231)

cia parallela e di due disk drive. Inoltre, svolgendo un'indagine tra le liste di sistema mantenute dall'Amiga abbiamo scoperto l'esistenza di diversi task (usualmente in attesa) che si occupano del refresh degli schermi MS-DOS e dello scambio di dati tra i due sistemi.

Va segnalato che nonostante la presenza della scheda A2620 non vi sono stati problemi di nessun tipo, tranne un'incertezza al momento dell'inizializzazione della bridgeboard. In pratica accadeva che accendendo l'Amiga o resettandolo, nel 20 per cento dei casi l'Amiga non riconosceva la scheda PC, e quando poi si tentava di accedervi mostrava il tipico messaggio che appare quando una scheda non funziona ("Library not found"). In questi casi, abbiamo sempre dovuto riaccendere o resettare nuovamente la macchina. A questo proposito è interessante notare che anche quando non siamo riusciti ad accedere alla bridgeboard, abbiamo comunque riscontrato che questa aveva completato perfettamente la propria procedura di startup (visibile sul secondo monitor di cui disponevamo durante la prova), per quanto dal lato Amiga la presenza della scheda A2620 non avesse permesso a BINDDRIVERS il caricamento della libreria di gestione janus.library. Non conosciamo le ragioni di questo estemporaneo malfunzionamento che si verifica quando sull'Amiga è montato il 68020, ma pensiamo sia da imputare a problemi di temporizzazione. Comunque, una volta concluso correttamente l'avviamento, la presenza della scheda A2620 non ha più causato problemi, e l'unica differenza rispetto al 68000 era il lieve miglioramento (in velocità) della gestione degli schermi simulati. Quindi, per aumentare la velocità e la fluidità di scroll degli schermi virtuali le strade da seguire sembrano essere essenzialmente due: l'uso di una scheda A2620 o la riduzione del numero di bit-plane usati nello schermo PC.

Dal punto di vista dell'affidabilità e della robustezza possiamo tranquillizzare i potenziali acquirenti: tutto funziona molto bene dal lato PC, per quanto riguarda la compatibilità MS-DOS e la velocità di calcolo, e anche dal lato Amiga i risultati sono positivi. In ogni caso gli sforzi effettuati per scrivere un affidabile software di comunicazione sono encomiabili.

Ci spiace invece che nel kit A2286 il connettore offerto sia predisposto per collegare in cascata solo due disk drive da 5,25". Non sarebbe stato difficile includere un connettore un po' più sofisticato che permettesse di collegare come secondo disk drive MS-DOS il disk drive da 3,5" di cui l'utente può disporre nell'Amiga... magari con un interruttore che prima dell'accensione permetta di scegliere se il disk drive aggiuntivo da 3,5" dev'essere visto dall'AmigaDOS o dall'MS-DOS. Comunque, con l'aiuto di un tecnico non è difficile costruirsi uno di questi connettori, aumentando così le possibili configurazioni di memorie di massa realizzabili con l'Amiga e la scheda bridgeboard.

Conclusioni

Tenendo conto anche del rapporto prezzo/prestazioni, possiamo dire che siamo stati favorevolmente impressionati in particolare dalla scheda A2286. Questo emulatore mette infatti a disposizione un completo computer IBM AT a un prezzo piuttosto basso e con prestazioni notevolissime. La scheda XT offre invece qualcosa di meno rispetto al suo prezzo, e in assoluto offre prestazioni che nell'ottica della tecnologia attuale sono senz'altro scadenti. Non è ovviamente un problema di emulazione (che è stata realizzata con la massima efficienza) ma è un problema del computer emulato. Potrebbe forse essere una buona idea per la Commodore immettere sul mercato una nuova versione della A2088 con clock a 8 MHz.

Quindi, a meno che non si voglia risparmiare a tutti i costi, a nostro avviso la scheda A2286 rappresenta una soluzione più che valida per chi desidera utilizzare il software MS-DOS sul suo Amiga 2000. ■

AT A2286

Specifiche:
Scheda bridgeboard da installare a cavallo di uno slot Amiga e uno IBM PC (rende possibile l'esecuzione di programmi scritti per la CPU 8086/8088/80286 sull'Amiga). Coprocessore matematico 80287 opzionale
Microprocessore:
Intel 80286 funzionante a 8 MHz
Memoria RAM:
1024K su scheda, espandibile tramite schede aggiuntive sul bus PC
BIOS:
BIOS Commodore, compatibile con il BIOS IBM AT; è autoconfigurante e contiene una routine di setup per facilitare l'installazione di nuove periferiche
Grafica:
Dall'Amiga vengono emulati i modi grafici tipici delle schede MDA e CGA per IBM PC
Memorie di massa:
Un disk drive da 5,25", che lavora con dischi formattati a 360K, 720K o 1,2 MB
Controller:
In grado di gestire due disk drive interni, da 3,5" o 5,25", con dischi di capacità compresa tra 360K e 1,2 MB (5,25") o tra 360K e 1,44 MB (3,5")
Espandibilità:
Sul bus PC dell'Amiga restano liberi tre slot, di cui uno a 16 bit compatibile AT
Dispositivi:
Controller per due disk drive.
Porta parallela Centronics emulata sull'analoga porta Amiga.
Orologio con batteria tampone.
Interfaccia per mouse Microsoft o compatibili emulata tramite una delle due porte mouse dell'Amiga
Tastiera:
Tastiera IBM AT emulata con la tastiera Amiga
Dimensioni:
Formato standard delle schede per Amiga 2000
Prezzo al pubblico:
L. 1.985.000 + Iva
Distribuzione:
Commodore Italiana spa
Viale Fulvio Testi, 280
20126 Milano (Tel. 02/661231)

I PACCHETTI GEOS 2.0 E GEOS 128 2.0

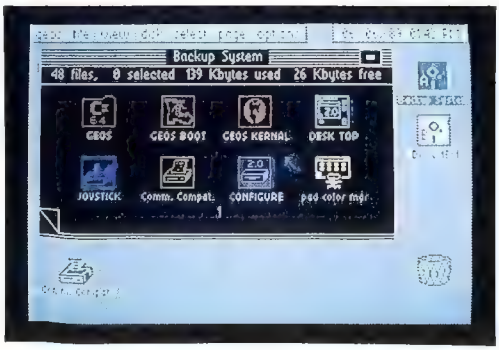
Tra i miglioramenti della versione 2.0 c'è la possibilità di gestire tre disk drive, compreso il 1581. GeoPaint finalmente prevede alcuni tool di geoPublish, come la capacità di distorcere i disegni

di Luca Giachino

GEOS 2.0 e *GEOS 128 2.0* costano rispettivamente 95.000 lire e 110.000 lire: sono certamente tra i pacchetti più cari disponibili per queste due macchine, ma il prezzo va sempre valutato in rapporto alle prestazioni fornite e a questo proposito vi sono alcune considerazioni da fare.

Al costo di un solo pacchetto applicativo l'ultima versione di *GEOS*, la 2.0, offre un rinnovato programma di disegno, un word processor WYSIWYG grafico accompagnato da diverse fonti-carattere, un completo set di driver di stampa e cinque driver di input, uno spell-checker, un driver per la stampante Apple LaserWriter, un'applicazione per importare file creati con altri word processor, un integratore di nominativi all'interno di testi redatti con il word processor, una serie di utility e desk accessory... e infine l'elemento principale: un sistema operativo user-friendly del tutto simile a quello dei famosi Macintosh. E ancora non è tutto.

GEOS è un sistema operativo aperto, all'interno del quale i più intraprendenti possono sviluppare applicazioni dotate d'interfacce grafiche dalle elevate prestazioni, un sistema in cui il kernel fa davvero miracoli per ridurre il carico di lavoro dei programmatori. Se tutto questo dovesse sembrare ancora poco, non dimentichiamo la raffinatezza e la cura con cui la Berkeley presenta sempre i suoi prodotti, accompagnati da manuali che difficilmente trovano eguali perfino nel software dell'Amiga. Infine, e questa è forse la considerazione più interessante, acquistando il sistema operativo *GEOS* si entra a far parte di un mondo in continua evoluzione, che si rinnova e che offre sempre nuovi aggiornamenti e applicazioni, un mondo che non abbandona i suoi utenti: chi ha acquistato *GEOS* può essere sicuro che alla Berkeley Softworks continueranno a lavorare per lui, una sensazione che nel mondo del software non si prova spesso. Certo, se guardassimo alla versione 2.0 come a un comune pacchetto software, 110.000 lire (versione per il C-128) potrebbero apparire eccessive, ma dal punto di vista che abbiamo illustrato possiamo solo concludere che si tratta di un prodotto caratterizzato da un rapporto prezzo/prestazioni eccezionalmente buono.



La versione 2.0 di deskTop prevede icone colorate e orologio

Prima di affrontare l'analisi della nuova versione 2.0, premettiamo che la maggior parte delle

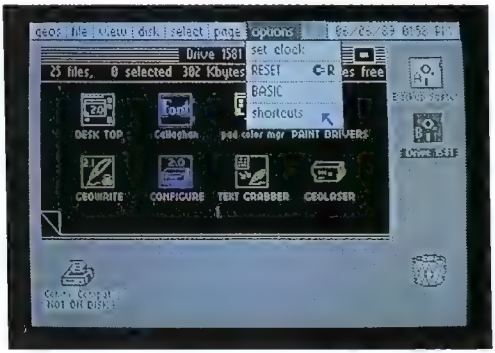
considerazioni che faremo nella prima parte di questo articolo valgono sia per *GEOS 2.0* sia per *GEOS 128 2.0*. Nella seconda parte descriveremo invece le caratteristiche fornite dal solo *GEOS 128 2.0*.

La versione 2.0

L'attesa versione 2.0 di *GEOS* costituisce un importante passo avanti sotto diversi profili. Anzitutto, il pacchetto si presenta in una nuova formula che riunisce un numero di applicazioni molto superiore a quello delle precedenti versioni: oltre al sistema operativo e alle sue applicazioni di contorno (*deskTop*, *geoPaint*, i desk accessory, i driver di stampa...), nella stessa confezione sono inclusi l'intero pacchetto *geoWrite Workshop* (*geoWrite 2.1*, *Text Grabber*, *geoLaser*, *geoMerge* e *Paint Drivers*) e l'applicazione *geoSpell*. Si tratta di una scelta felice, perché mette subito a disposizione dell'utente tutti gli strumenti necessari per il word processing, per la gestione e la correzione dei testi, e per il disegno: una base di partenza decisamente più completa che nel passato.

GEOS 2.0 è quindi un pacchetto piuttosto voluminoso, che prevede quattro dischi (uno è il solito disco per il collegamento con la rete Q-Link, relativamente inutile per il mercato italiano), un manuale da 300 pagine, un foglio trasparente con una griglia disegnata da posare sui disegni da ricalcare (come riferimento nell'uso di *geoPaint*), e per il C-128 un manuale supplementare di cento pagine che integra il manuale principale, identico per entrambe le versioni.

Il manuale sul sistema operativo stupisce per la sua completezza, per la ricchezza di dettagli e per l'ottima organizzazione degli argomenti. Come al solito prevede una parte dedicata all'installazione, un tutorial per sperimentare rapidamente le peculiarità del sistema, una serie di capitoli dedicati alle varie applicazioni, compresi i desk accessory, cinque appendici su svariati argomenti di utilità generale, un glossario e infine un indispensabile indice analitico di sette pagine. Inoltre, sebbene sia in inglese è scritto in maniera molto chiara ed è ricco d'immagini che aiutano notevolmente la lettura.



Il disk drive 1581 è finalmente utilizzabile da GEOS 2.0

Il disco 1 contiene sulla prima facciata il sistema operativo protetto, i file che permettono il boot e le applicazioni di contorno, mentre sulla seconda facciata risiedono cinque interessanti demo *GEOS* compatibili che mostrano in modo simpatico e intuitivo le peculiarità dei pacchetti *geoFile*, *Fontpack Plus*, *geoPublish*, *Deskpack Plus* e *geoCalc*. Si tratta ovviamente di pubblicità, e qualcuno potrebbe non gradirla, ma si rivela senz'altro utile per valutare con un rapido colpo d'occhio gli altri pacchetti prodotti dalla Berkeley.

Sul secondo disco risiede il consueto backup del disco sistema, le applicazioni *geoWrite* e *geoPaint* insieme con i desk accessory Photo Manager, Calculator, e Note Pad, e alle sei fonti-carattere standard (una è la fonte Commodore non proporzionale). Sul terzo disco, infine, risiedono tutte le utility per la gestione dei testi (*geoLaser*, *geoMerge*, *Text Grabber* e i relativi file di supporto), le fonti-carattere che riproducono a schermo le fonti residenti disponibili sulla stampante laser Apple LaserWriter, e l'applicazione *geoSpell*.

Il kernel di GEOS 2.0

Dal punto di vista dell'utente, il nuovo kernel, che è perfettamente compatibile con le precedenti versioni, aggiunge rispetto alla 1.3 per il C-64 la capacità di accedere non solo ai disk drive 1541 e 1571 e alle espansioni RAM, ma anche al 1581, il disk drive Commodore nato per i dischi da 3,5" per un totale di 790K utili. Si tratta di un'importante novità che era già disponibile per gli utenti di *GEOS 128* da parecchio tempo. La capacità dei dischi da 3,5" e la più elevata velocità d'accesso del 1581 sono infatti due caratteristiche che ben si sposano con le esigenze di spazio e rapidità negli accessi tipiche di *GEOS*. Su un singolo disco da 3,5", per esempio, può comodamente risiedere tutto il contenuto dei tre dischi (escludendo la facciata con i demo). Peccato solo che non sia stata prevista la possibilità di effettuare il boot dal 1581 (probabilmente perché avrebbe reso molto semplice copiare il disco sistema protetto).

Dopo aver restituito vigore al C-64 e al C-128 e aver lanciato le espansioni RAM, con questa aggiunta *GEOS* riporta in auge il 1581, che come tutti sanno è un'ottima periferica ingiustamente penalizzata dalla mancanza di software adeguato. Come già è accaduto altre volte, la Berkeley si dimostra particolar-

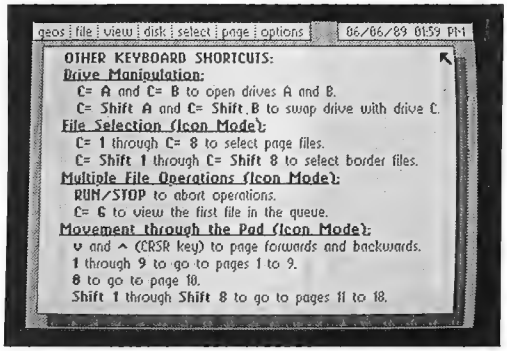
mente attenta al mondo Commodore e ai suoi utenti.

Un'analisi dei codici ha mostrato che le funzioni di accesso ai dischi sono state rimaneggiate per consentire l'uso dei driver di accesso ai disk drive, e vi sono altre differenze che comunque sono del tutto trasparenti dall'esterno: le applicazioni nate con la versione 1.3, infatti, hanno dimostrato di funzionare perfettamente, e la tavola di salto a $C000 contiene lo stesso numero di vettori della versione precedente.

In particolare il kernel è sempre in grado di "vedere" non più di due disk drive. Se, come vedremo fra poco, si riesce a interagire con tre disk drive è solo grazie a un particolare accorgimento che non ha a che fare con il sistema operativo.

Per quanto riguarda l'installazione del sistema, ricordiamo che ogni kernel prodotto dalla Berkeley dispone di un proprio numero di serie che lo contraddistingue. Le applicazioni per le quali la Berkeley non consente la libera distribuzione, come *geoWrite V2.1* e *geoPublish*, memorizzano al loro interno questo numero di serie la prima volta che vengono mandate in esecuzione, e in seguito si rifiutano di funzionare con un kernel che ne possieda uno diverso. Ma gli utenti che possiedono già queste applicazioni e le hanno installate con la precedente versione del sistema operativo possono stare tranquilli: *GEOS 2.0*, infatti, al momento dell'installazione interroga l'utente per sapere se è un nuovo utente *GEOS* oppure no. Nel secondo caso, richiede d'inserire un disco contenente una qualunque di queste applicazioni, dalla quale attinge una misteriosa "chiave" e la memorizza al proprio interno come numero di serie. In questo modo, se l'utente ha seguito l'intera procedura d'installazione, disporrà di un kernel perfettamente compatibile con tutte le applicazioni protette di cui già dispone. Se invece l'utente non ha mai impiegato *GEOS*, il kernel si installa memorizzando al suo interno un nuovo numero di serie.

È importante non commettere errori in questa fase, perché una volta installato il sistema non si può tornare indietro. Si tenga inoltre presente che se l'utente dispone del C-128 e ha sempre impiegato *GEOS* con il C-64, può installare *GEOS 128 2.0* con lo stesso numero di serie di *GEOS 64*.



Il memorandum sulle combinazioni di tasti disponibili

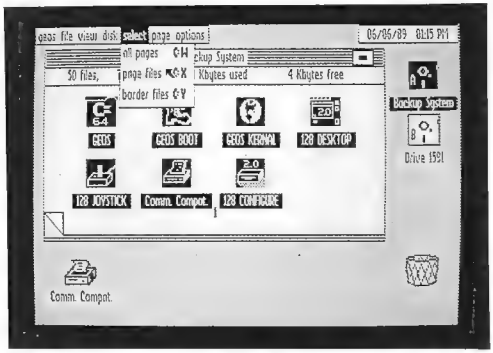
DeskTop V2.0

Questa è indubbiamente l'applicazione che nel passaggio alla versione 2.0 del sistema ha subito la maggior quantità di modifiche. Appena il consueto bloc-notes appare sullo schermo saltano subito all'occhio alcune differenze sostanziali. In alto a sinistra appare finalmente un orologio dotato di data. Non ci dilunghiamo su quanto sia stata criticata l'assenza di un orologio all'interno di *deskTop*, e ci limitiamo a salutare con approvazione la novità. L'orologio è facilmente regolabile con il mouse o tramite la voce Set Clock del menu Options. Un altro particolare degno di nota è l'icona della stampante situata in basso a sinistra, finalmente lontana da quella del cestino. Segnaliamo poi l'aggiunta di due voci nel menu principale, Select (per selezionare tutti i file del disco, i file contenuti nella pagina visibile, o solo quelli depositati sul bordo) e Page (per aggiungere e cancellare rapidamente pagine di directory). Al momento della cancellazione si noterà che il nome dell'ultimo file cancellato appare sotto il cestino, per segnalare che è recuperabile tramite la nuova voce Undo Delete del menu File. Altre novità sono la frase NOT ON DISK sotto l'icona della stampante qualora sul disco aperto non risulti presente il driver di stampa prescelto e un'indicatore del numero di file selezionati nell'intestazione del bloc-notes.

A un più attento esame dei menu notiamo la presenza di un nuovo desk accessory, Pad Color Mgr, che permette di variare il colore di fondo del bloc-notes e attribuire un colore diverso a ogni tipo di file previsto da *GEOS*, colore che assumeranno tutte le icone relative ai file di quel tipo. Nel menu Disk appare ora la voce Erase, che permette la formattazione software di un disco. Come dovrebbero sapere gli utenti dei disk drive Commodore, oltre a formattare fisicamente un disco, cioè creare tutti quei riferimenti sul supporto magnetico che lo suddividono in tracce e settori cancellandone ogni contenuto, il DOS consente anche di effettuare una formattazione software che consiste unicamente nella cancellazione dell'intera directory e nell'azzeramento della BAM, in modo che il disco sia riutilizzabile, come se nessun file vi fosse mai stato memorizzato. Rispetto alla formattazione fisica, questa procedura è ovviamente molto più rapida.

Le selezioni multiple, la gestione delle pagine

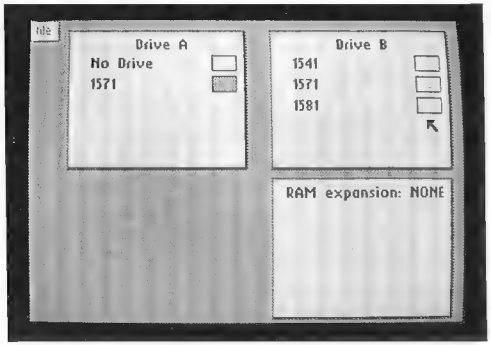
Rispetto al passato, la novità più interessante è sicuramente quella di poter selezionare diversi file contemporaneamente ed effettuare su di essi operazioni di copia, cancellazione e alterazione del nome, che in precedenza potevamo eseguire solo su un file alla volta. Si tratta di un'importante e utile miglioria. Ora, agendo con il menu Select o direttamente con la tastiera e il mouse, l'utente può selezionare più file di una stessa pagina, più file disposti su diverse pagine, alcuni o tutti i file disposti sul bordo e perfino tutti i file presenti su disco, e poi copiarli su un altro disco muovendo semplicemente una particolare icona fantasma che "si attacca" al puntatore sullo schermo e rappresenta tutti i file selezionati. Con una semplicissima operazione, l'utente riesce così a copiare diversi file, operazione che in precedenza era abbastanza lunga e noiosa. Oltre alla copia, le selezioni multiple permettono la cancellazione simultanea di un gruppo di file, o l'alterazione dei loro nomi uno dopo l'altro in sequenza. Sebbene manchi ancora la possibilità di redigere file comandi atti ad automatizzare le procedure ripetitive, la selezione multipla dei file costituisce già un ottimo passo avanti, e si rivela davvero utile in moltissime occasioni. Nella gestione dei file e dei dischi, *deskTop V2.0* offre anche alcune nuove funzioni che permettono di gestire meglio le pagine delle directory. Insieme alla selezione multipla, rendono più semplice il riordinamento dei dischi. Nella sua globalità, *deskTop* è anche migliorato sotto altri aspetti. Per esempio, scegliendo un nuovo driver di stampa o di input, *deskTop* si preoccupa di trasformare il driver prescelto nel driver di default, in modo che quando si riattiva il sistema l'operazione non deve più essere ripetuta. Per quanto riguarda i driver di stampa, il pacchetto contiene ora ben 30 driver che, secondo quanto si dice nel manuale, sono stati provati con 59 tipi diversi di stampanti (30 driver di stampa non li possiede neanche il disco *Workbench 1.3* dell'Amiga!). Anche le combinazioni di tasti di cui l'utente dispone per evitare l'uso del mouse sono aumentate, al punto che ora *deskTop* permette di visualizzare un box nel quale le riassume tutte, permettendo all'utente una consultazione immediata. Per esempio, ora le icone dei file situate sul bloc-notes o sul bordo possono essere selezionate con la tastiera (anche in gruppo). Sempre con la tastiera si possono scorrere tutte le 18 pagine della directory.



Ora deskTop consente di effettuare selezioni multiple di file

Il sistema dei dischi

Tramite il nuovo programma CONFIGURE 2.0 l'utente può scegliere fra diverse configurazioni del sistema a seconda dell'eventuale presenza di un'espansione RAM e di due o più disk drive. Le differenze sostanziali rispetto alle versioni precedenti sono che ora si può assegnare un disk drive C, oltre all'A e al B, e che si può impiegare il disk drive 1581. Per essere più precisi, se si dispone almeno dell'espansione RAM 1764 (256K), è possibile abilitare tre disk drive, uno dei quali dev'essere necessariamente un RAM disk. Nel caso che l'utente abbia scelto la configurazione con tre disk drive, nell'uso saranno sempre e solo due i disk drive utilizzabili, mentre un terzo rimarrà disattivato fino a quando, tramite *deskTop*, non verrà scambiato con uno degli altri due. In pratica, configurando il sistema con tre disk drive, sullo schermo



Da CONFIGURE 2.0 si indicano i disk drive di cui si dispone

di *deskTop* appaiono sulla destra tre icone per i disk drive, delle quali le due più in alto corrispondono ai disk drive attivi, mentre la terza individua quello non attivo. L'interazione con i primi due disk drive avviene come di consueto, mentre se si desidera accedere al terzo disk drive basta effettuare una semplice procedura con il mouse. Dal punto di vista delle applicazioni il terzo disk drive è come se non esistesse.

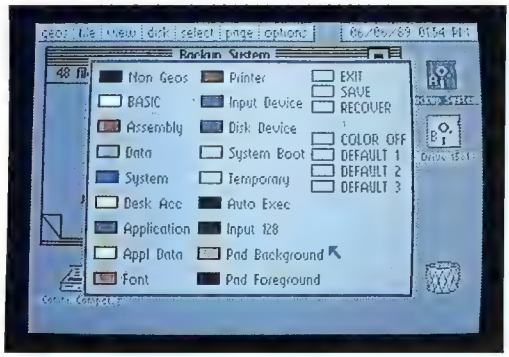
Come abbiamo già spiegato, questa macchinosa gestione è dovuta al fatto che il kernel non è in grado di accedere a più di due disk drive; per superare questo limite è stato giocoforza ricorrere a una piccola "magia" un po' scomoda.

Sebbene in questo modo si possano impiegare ben tre memorie di massa, e nonostante una di queste possa essere finalmente il 1581, l'operazione che l'utente deve effettuare dall'interno di *deskTop* per scambiare i disk drive poteva essere resa del tutto trasparente, senza obbligare a una procedura di scambio. In ogni caso, i vantaggi sono decisamente apprezzabili, anche se gli utenti che possiedono più di un disk drive non sono molti. Si noti che sul disco non appaiono driver di accesso ai disk drive, dal momento che sono tutti contenuti nel file CONFIGURE. Oltre alla gestione dei tre disk drive, CONFIGURE 2.0 permette di attivare le consuete opzioni per il reboot dall'espansione RAM e per lo spostamento veloce di blocchi di memoria impiegando il DMA (Direct Memory Access) dell'espansione.

Per concludere, bisogna ricordare che se non si dispone di un'espansione RAM occorre che il file CONFIGURE 2.0 segua sempre l'applicazione *deskTop* che l'utente memorizza nei vari dischi di lavoro, in quanto *deskTop*, ogni volta che viene eseguita, accede al file CONFIGURE per sapere qual è la configurazione attuale. Peccato che nel kernel non abbiano previsto uno spazio per salvare permanentemente i dati relativi alla configurazione preferita.

DeskTop a colori

L'ultima peculiarità di *deskTop* da segnalare è la possibilità di colorare le icone dei file e il bloc-notes su cui sono collocate. In più, il desk accessory Pad Color Mgr permette di assegnare un colore a ogni tipo di file, in modo che icone relative a file dello stesso tipo vengano contraddistinte dallo stesso colore. Il sistema è utile soprattutto per distinguere tra i file eseguibili e quelli non eseguibili, come le fonti-carattere, i driver, e i file di sistema.



Il desk accessory Pad Color Mgr per i colori delle icone

Attivando questo desk accessory, appare una finestra nella quale a ogni tipo di file corrisponde un rettangolo. Agendo su ogni rettangolo l'utente può scorrere i 16 colori disponibili con l'hardware del C-64 e del C-128 (per questa seconda macchina è opportuno ricordare che nel modo a 80 colonne i colori non vengono generati).

Per facilitare le scelte cromatiche, il desk accessory offre anche tre configurazioni di default di possibili colori. La configurazione prescelta può essere semplicemente attivata oppure anche salvata all'interno di un file di nome Pad Color Pref.

Perché i colori prestabiliti appaiano su ogni disco dotato di *deskTop*, è opportuno che l'utente vi copi questo file che *deskTop* cerca e consulta ogni volta che va in esecuzione.

GeoPaint

Nel mondo di *GEOS*, *geoPaint* era l'applicazione che finora aveva subito meno migliorie, ma non perché non ne avesse bisogno. In effetti forse è stata un po' trascurata in questi anni, a beneficio di applicazioni come *geoPublish*. Ma, finalmente, con *GEOS 2.0* gli appassionati di disegno dispongono di nuovi strumenti.

È ora possibile muovere il puntatore sullo schermo impiegando i tasti cursore per effettuare spostamenti di precisione. Nel disegno di segmenti si può disegnare una spezzata, e costringere i segmenti ad assumere angolazioni multiple di 45 gradi.

Anche il disegno dei rettangoli e dei cerchi è stato migliorato. Nella precedente versione risultava difficile disegnare un quadrato ed era impossibile disegnare ellissi. Ora, invece, possiamo forzare il tool di disegno dei rettangoli a visualizzare quadrati, mentre il tool di disegno dei cerchi disegna ellissi per default; per disegnare cerchi bisogna forzarlo. Peccato soltanto che se un cerchio esce dai limiti della finestra *geoPaint* continua a ridisegnarlo per indicare che non è interamente visibile.

Un'altra interessante novità riguarda la selezione delle aree. Nella nuova versione, il tool di selezione delle aree funziona esattamente come in *geoPublish* (per la verità, ci eravamo sempre chiesti perché la Berkeley non si

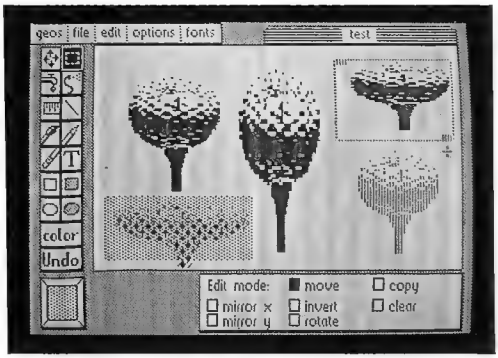
decideva a dotare anche *geoPaint* di alcune importanti funzioni nate per *geoPublish*). Quando sono state decise le dimensioni del box tratteggiato che delimita il perimetro dell'area, appaiono due gadget su due vertici opposti che permettono di muovere il box e cambiarne le dimensioni prima che diventi definitivo. Si tratta di un sistema veramente comodo per la selezione delle aree, ma anche per importare un file scrap tramite le opzioni di cut & paste.

Sempre da *geoPublish* derivano le routine che permettono di effettuare particolari operazioni sulle immagini che vengono importate. Nell'ordine, queste operazioni consentono di centrare l'immagine all'interno dell'area selezionata, di cambiarne le dimensioni in modo proporzionale perché occupi per intero un lato dell'area selezionata, di cambiarne le dimensioni in modo non proporzionale perché si estenda fino a coprire entrambi i lati dell'area selezionata, di scegliere fra 32 matrici grafiche, di abilitare o meno lo smussamento delle scalettature sui contorni e infine di abilitare la trasparenza. L'aggiunta di queste opzioni costituisce un vero salto di qualità per *geoPaint*.

Ora è anche possibile attivare una griglia nella quale i vertici distano fra loro un pollice esatto. Per quanto utile, questa griglia presenta due limiti: il passo fra i vertici non può essere alterato, e quando viene disattivata cancella tutto quello che è stato disegnato o che appariva sotto le sue linee.

La funzione di riempimento di aree si può ora bloccare, evitando così lunghi tempi di attesa quando si commette un errore. Anche la funzione Spray è stata migliorata, permettendo di scegliere fra il comportamento normale, la visualizzazione in negativo della matrice grafica e la visualizzazione dei soli punti neri o dei soli punti bianchi.

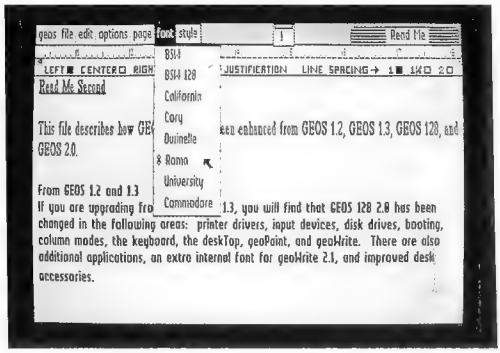
Sebbene a nostro avviso *geoPaint* abbia ancora bisogno di molte migliorie, la nuova versione che accompagna *GEOS 2.0* costituisce già un risultato di buon livello. Speriamo che nel futuro la Berkeley dedichi a questa importante applicazione tutta l'attenzione che merita. Per quanto riguarda *geoPaint 128*, rimandiamo il lettore all'ultima parte di questo articolo.



GeoPaint consente di ottenere interessanti effetti con i disegni

Altre novità

Delle restanti applicazioni contenute nel pacchetto, *geoWrite V2.1*, *geoLaser* e *Paint Drivers* sono immutate rispetto alla precedente versione di *GEOS* e al pacchetto *geoWrite Workshop*. Calculator consente ora di effettuare il cut dell'ultimo risultato ottenuto, un'opzione abbastanza inutile, dal momento che non è difficile tenere a mente un risultato di dieci cifre. Oltre a questo, è fastidioso dover impiegare una combinazione di tasti anziché il consueto menu Edit. La stessa osservazione vale per Note Pad, che consente ora di effettuare operazioni di cut & paste ma senza ricorrere ai menu.



GeoWrite 128 mostra uno schermo davvero professionale

GeoMerge è stato aggiornato alla versione 2.1, ma si tratta di una revisione interna che non porta sostanziali novità. *Text Grabber* consente ora d'importare file redatti da un maggior ventaglio di word processor, mentre *geoSpell* è stato migliorato sotto il profilo della velocità. Per maggiori dettagli su queste applicazioni vi rimandiamo ai vari articoli dedicati al mondo di *GEOS* apparsi sui precedenti numeri di *Commodore Gazette*.

Qualche parola in più la meritano invece alcuni desk accessory.

I desk accessory Photo Manager e Text Manager consentono ora di assegnare nomi rispettiva-

mente alle immagini e ai testi così da permettere ricerche random più veloci della scansione sequenziale.

GEOS 128 2.0

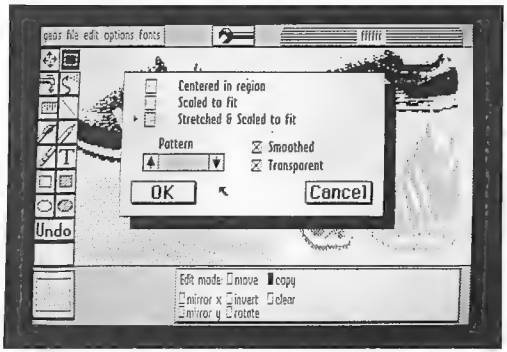
Se per il C-64 l'avvento della versione 2.0 di *GEOS* costituisce una novità di rilievo, per il C-128 è un deciso passo in avanti. Tutte le considerazione fatte finora valgono anche per questo sistema operativo, ma se ne aggiungono alcune altre ampiamente descritte dal manuale integrativo di cento pagine incluso nel pacchetto.

Com'è ormai noto, una delle fondamentali differenze della versione per C-128 è che consente l'uso dello schermo monocromatico a 80 colonne, per una risoluzione di 640 x 200 pixel. Il risultato è uno schermo che orizzontalmente ha una risoluzione maggiore di quella del Macintosh e offre un'immagine estremamente nitida. *GEOS 128* ne trae ovviamente ottimi vantaggi: in particolare *geoWrite* (che funziona soltanto a 80 colonne) e *geoPaint*, che grazie alla maggiore risoluzione orizzontale consentono ora di evitare lo spostamento orizzontale della finestra sulla pagina; un grande vantaggio rispetto alla versione per il C-64.

Le altre differenze della versione per il C-128 riguardano i driver di stampa, per alcuni dei quali *GEOS 128* aggiunge le versioni "DS" (double-strike), "QS" (quadruple-strike), e "RED" (che compattano la pagina permettendo di ottenere immagini più nitide).

In generale, la versione per il C-128 di *deskTop* aggiunge molti controlli da tastiera che possiamo schematicamente ordinare nei seguenti punti: controlli per la selezione dei disk drive, per la copiatura e gli spostamenti dei file, per la selezione sequenziale dei file, per sfogliare le pagine della directory e infine per aprire i menu a tutti i livelli e selezionarne le voci senza impiegare il mouse.

GeoPaint può funzionare sia a 40 colonne sia a 80. A parte questa differenza e la possibilità di far scomparire le icone dei tool di disegno, quest'applicazione è esattamente uguale alla versione per il C-64. Il word processor *geoWrite 128* funziona solo a 80 colonne, e in questo modo grafico



Le opzioni che geoPaint offre per alterare i nostri disegni...

si può realmente definire WYSIWYG, in quanto gli otto pollici e mezzo disponibili in larghezza durante la stampa occupano l'intera larghezza di schermo. Una seconda differenza rispetto alla versione per il C-64 è una nuova fonte-carattere residente di cui

...e i bizzarri, ma anche utili risultati che si possono ottenere

dispone *geoWrite 128*. Anche *geoSpell* funziona solo in 80 colonne. *Text Grabber* è in grado d'importare testi prodotti con alcuni word processor specifici per il C-128, come *WordWriter 128*. Sostanzialmente, queste sono le uniche differenze fra i due pacchetti.

Conclusioni

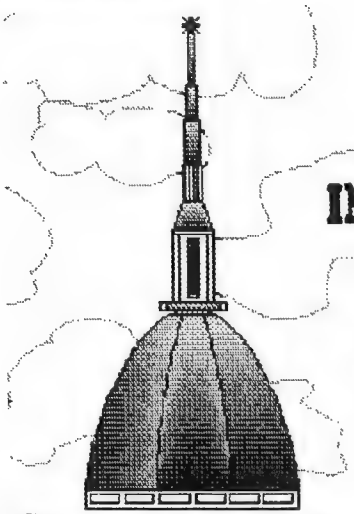
Come si sarà già intuito dall'analisi fin qui svolta, il giudizio nei confronti della neonata versione 2.0 di *GEOS* non può che essere positivo per entrambi i computer. Le migliorie che riguardano *deskTop* e *geoPaint* sono le più interessanti e contribuiscono a rendere sempre più completo questo sistema operativo.

Il mondo di *GEOS*, a quanto sembra, non ha affatto smesso di crescere e di fornire ai suoi utenti strumenti sempre migliori. ■

I pacchetti GEOS sono disponibili presso:

Lago snc
(GEOS 2.0, L. 95.000,
GEOS 128 2.0 L. 110.000)
Via Napoleona, 16
22100 Como
(Tel. 031/300174)

informatica ITALIA®



da TORINO in tutto il paese

IMPORTA e SVILUPPA la Tecnologia della

COMPUTER GRAFICA
COMPUTER MUSICA

Su computer Commodore AMIGA:

*Telecamere e videoregistratori
*Digitalizzatori:
→VIDEON a colori con encoder
→VD Amiga in tempo reale
→FRAMER in tempo reale a colori con encoder e memorie di quadro
*Genlock: → amatoriale PAL
→ professionale PAL Broadcast
RENDALE-VIDEOCOMP-NERIKI-MAGNI
*Software ORIGINALE per Computer Grafica
*Stampanti a colori a getto d'inchiostro
XEROX 4020
*Applicazioni musicali:
→accessori e software

INFORMATICA ITALIA S.r.l.
C.so Re Umberto, 128
10128 TORINO Tel. 011/501647

LA MPS 1230 STUPISCE SOLO PER IL PREZZO: TROPPO ELEVATO

Una nuova stampante Commodore per C-64/128 e Amiga che presenta ben poche attrattive: le caratteristiche essenziali sono scarsa affidabilità e prezzo più alto della media

di Luca Giachino

Talvolta esaminando certi prodotti abbiamo l'impressione che siano stati lanciati sul mercato, è proprio il caso di dirlo, senza prendere la mira. In altre parole, non s'intravede una benché minima strategia di marketing per agevolarne l'affermazione, per rendere il prodotto accattivante e fronteggiare la concorrenza. Eppure le strategie di mercato più elementari dovrebbero essere ovvie per chiunque: una scelta di qualità superiore, l'introduzione di potenzialità "mirate" a una particolare fascia d'utenza e, nei casi peggiori, una politica di prezzi che riesca a mettere in secondo piano prestazioni mediocri.

È il caso della Commodore MPS 1230, una stampante per l'Amiga, i PC e per il C-64/128 che non introduce nessuna vera novità rispetto a prodotti analoghi e presenta caratteristiche tanto mediocri che potrebbe distinguersi positivamente solo con un prezzo molto concorrenziale. Quando abbiamo recensito la MPS 1500C, una stampante simile da un punto di vista costruttivo, abbiamo espresso una valutazione favorevole perché offriva una buona stampa a colori a un prezzo abbordabile. Per fare un altro confronto, la Mannesmann MT81 (di cui pubblicheremo la recensione nel prossimo numero) è anch'essa una periferica collegabile sia ai PC sia agli 8 bit della Commodore, ma offre prestazioni paragonabili a quelle della MPS 1230 a un prezzo inferiore.

Un'altra considerazione riguarda la doppia interfaccia che ne consente il collegamento a tutte le macchine Commodore. La scelta si dimostra positiva solo per coloro che possiedono un C-64/128: un domani, infatti, potrebbero acquistare l'Amiga senza cambiare stampante. Ci sembra al contrario improbabile che un utente dell'Amiga, per quanto nostalgico, possa decidere di fare dietrofront e di tornare agli otto bit. In altre parole l'interfaccia seriale per gli otto bit dovrebbe essere opzionale.

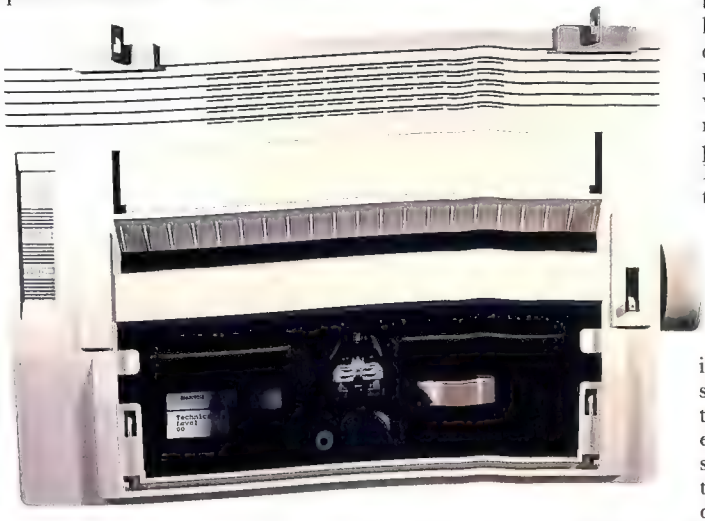
Uno sguardo d'insieme

La MPS 1230 è una stampante a impatto con una matrice di punti composta da nove aghi. A 10 cpi riesce a stampare 80 caratteri per riga a una velocità di 120 caratteri al secondo.

Dal punto di vista estetico è caratterizzata da linee geometriche non proprio essenziali che danno come risultano un insieme abbastanza gradevole, certamente migliore rispetto alla MPS 1500C. Nell'ormai classico beige Commodore, la MPS 1230 è di dimensioni contenute (94 x 370 x 253 mm) ed essendo prevalentemente di plastica pesa soltanto 4,2 Kg.

Sul lato superiore troviamo uno sportello fumé che permette di accedere al carrello di trascinamento della testina per sostituire il caricatore d'inchiostro. Lo stesso sportello svolge la funzione di premi-carta sul rullo di stampa, e assorbe il rumore prodotto dalla testina, peraltro già abbastanza contenuto. Ai due estremi del rullo risiedono due coperchietti a incastro che si devono togliere per installare il trattore esterno in dotazione alla macchina. In corrispondenza del coperchietto di destra è situata una leva che permette di selezionare il tipo di trascinamento. Sulla sinistra risiede invece la console di comando, composta da tre tasti a membrana e tre led di controllo. Due piccole aperture retrostanti alla feritoia d'inserimento della carta consentono la presenza di un "introduttore" per la carta che può svolgere due funzioni: se il trascinamento è a frizione, l'introduttore, orientato verticalmente, permette un più agevole inserimento dei fogli singoli; se invece il trascinamento è a trattore, l'introduttore, disposto orizzontalmente, si trasforma in separatore fra il flusso del modulo continuo entrante e il flusso uscente. Infine, sul bordo superiore posteriore sono inseribili due ulteriori guide mobili il cui corretto posizionamento, come vedremo, è molto importante nel trascinamento a trattore dei moduli continui.

La manopola per l'avanzamento manuale della carta è situata sulla destra, mentre sul lato posteriore troviamo (da sinistra verso destra) l'interruttore d'accensione, il cavo di alimentazione solidale con l'unità, la presa DIN a 6 pin per collegare la stampante con il C-64 e il C-128, e infine il connettore dell'interfaccia parallela Centronics.



Le prime impressioni

Lo sportello fumé di protezione, insonorizzazione e pressione della carta permette un rapido accesso al caricatore dell'inchiostro, montato direttamente sul carrello della testina di stampa. Sostituirlo non presenta difficoltà, ma nutriamo alcuni dubbi sulle sue ridotte dimensioni anche perché il costruttore dichiara una vita di 800 mila caratteri, contro la cifra di un milione e 200 mila che vengono dichiarati abitualmente con i caricatori fissi. Per i costruttori questo tipo di caricatore è comodo per due motivi: è la soluzione più semplice per limitare le dimensioni della stampante senza troppi sforzi, e richiede un meccanismo di trascinamento del nastro inchiostrato più semplice di quello necessario con i caricatori fissi che occupano l'intera larghezza di stampa. Si tratta di due vantaggi che agli occhi del produttore rendono questo sistema molto interessante, ma nel caso della MPS 1230 hanno dato come risultato un caricatore davvero minuto, ovviamente di durata puttosto limitata. Riguardo al caricatore, citiamo un altro particolare negativo che per gli utenti non avrà naturalmente molta importanza, ma che si aggiunge agli altri: durante la breve esposizione della stampante ai riflettori del nostro studio fotografico, il caricatore si è completamente deformato per il calore della forte illuminazione, ed è la prima volta che si verifica un problema del genere. Evidentemente anche la qualità della plastica non è delle migliori.

Per quanto riguarda il premicarta, riteniamo che il classico sistema composto da rullini rotanti sul rullo di trascinamento e premuti da uno stelo metallico sia senz'altro preferibile al semplice sportello impiegato dalla MPS 1230, il quale non esercita alcuna pressione sul rullo di stampa e quindi non garantisce una perfetta aderenza della carta al rullo durante i passaggi della testina.

Anche il trattore (da inserire quando si rivela necessario) presenta alcuni svantaggi. Prima di tutto costringe sempre a sprecare il primo foglio di stampa, proprio perché è collocato dopo la testina per garantire l'estraibilità... un problema comune a molte stampanti Commodore. Inoltre, il sistema adottato prevede che il trascinamento a frizione sia utilizzabile soltanto quando il trattore esterno non è inserito negli incastri, obbligando così a frequenti rimozioni, che ovviamente potrebbero danneggiare i delicati incastri in plastica. Come se non bastasse, il trattore è anche assai impreciso, le ruote dentate non sono veramente bloccabili ma scorrono a fatica sull'asta a sezione quadrata, e inserirle negli incastri della stampante a volte si dimostra decisamente fastidioso. Se per i precedenti modelli che adottavano lo stesso sistema non ci siamo espressi con troppa severità è perché pensavamo si trattasse di una poco fortunata tecnica sperimentale. Quando lo stesso sistema è riapparso ai nostri occhi anche con questo più recente modello, abbiamo dovuto prendere atto del fatto che evidentemente qualcuno lo ritiene un ottimo metodo... per risparmiare. Va invece rilevato che altre stampanti appartenenti alla stessa fascia di mercato costano meno e possiedono il trattore fisso completamente integrato prima della testina di stampa. Ci sembra quindi il momento di mettere in rilievo i gravi limiti di un sistema che ricorda la precisione d'incastro del Lego: scomodo e poco professionale.

Questo sistema presenta un altro enorme difetto che non possiamo tralasciare: disallinea molto facilmente la carta dopo

Prova di stampa della Commodore MPS 1230

Qualità di stampa disponibili

Draft	Near Letter Quality
abcdefghijklm	abcdefghijklm
NOPQRSTUVWXYZ	NOPQRSTUVWXYZ
1234567890+-*/	1234567890+-*/
,.;:@#!$%&()=?^	,.;:@#!$%&()=?^

Stili di stampa disponibili

Draft	Near Letter Quality
Corsivo	---
Nero	**Nero**
Grassetto	**Grassetto**
Doppia passata	---
Proporzionale	Proporzionale
~~Sopralineato~~	~~Sopralineato~~
Apici	Apici
Pedici	Pedici

Dimensioni particolari del carattere

Draft	Near Letter Quality
Largh. doppia	Largh. doppia
10 caratteri per pollice	10 caratteri per pollice
12 caratteri per pollice	12 caratteri per pollice
15 caratteri per pollice	15 caratteri per pollice
17,1 caratteri per pollice	17,1 caratteri per pollice
20 caratteri per pollice	20 caratteri per pollice
24 caratteri per pollice	24 caratteri per pollice
30 caratteri per pollice	30 caratteri per pollice

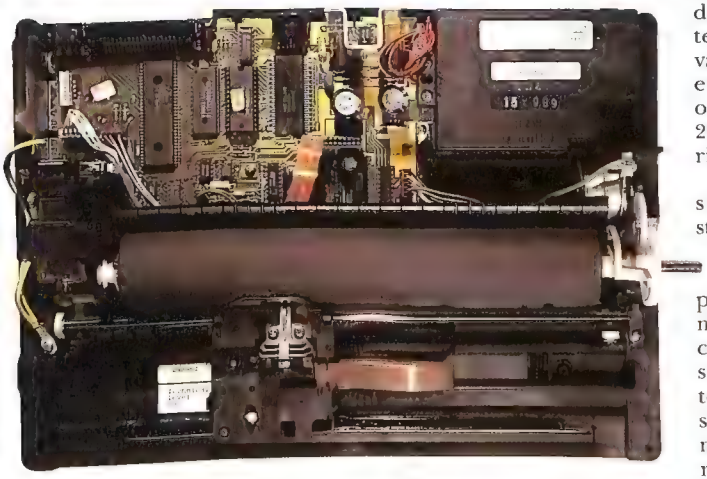
due o tre fogli stampati, se la carta in entrata non viene disposta più che bene, e il disallineamento peggiora sempre di più a mano a mano che la stampa prosegue, finendo per inceppare il meccanismo. A noi è sembrato un problema grave, perché in pratica obbliga a tenere sempre d'occhio la stampante. Un paio di volte, nel corso della prova, abbiamo avviato la stampa di testi piuttosto lunghi e ci siamo allontanati lasciando la macchina al lavoro... grave errore! Al ritorno abbiamo trovato regolarmente un vero e proprio disastro. Con i tradizionali trascinamenti a trattore questo problema non si è mai verificato.

Sorvolando sul fatto che occorrerebbe una scritta per indicare all'utente quali trascinamenti vengono selezionati dalle due posizioni della leva sulla destra del rullo, possiamo concludere questa prima serie d'impressioni sottolineando che l'introduttore orientabile può coesistere con il trattore esterno solo a patto che le due guide mobili vengano divelte facendo forza sui loro incastri, operazione che oltre ad aumentare ancora di più il disagio nel passaggio da un trascinamento all'altro può anche risolversi con la rottura degli incastri.

Il manuale, fortunatamente in italiano, è di facile consultazione, e costituisce una buona fonte d'informazioni sia per l'utente sia per il programmatore.

Le prestazioni

Per questa stampante il costruttore dichiara una velocità di 120 cps a 10 cpi in modo Draft bidirezionale. Come sempre accade, questa velocità massima non tiene conto del tempo necessario per l'avanzamento della carta fra una riga e l'altra, e quindi è un dato da valutare con attenzione. Dalla prova che abbiamo effettuato risulta che stampando righe di 80 caratteri (la quantità massima a 10 cpi), la MPS 1230 raggiunge in realtà i 75 cps, contro gli 80 cps della MPS 1500C. Si tratta di un dato più significativo di quello dichiarato dal costruttore in quanto è questa la reale velocità con la quale l'utente riesce a ottenere i propri stampati. La MPS 1230 dimostra che l'avanzamento carta diminuisce la velocità nominale di circa il 37 per cento, e non è quindi all'altezza della velocità di spostamento della testina, anche se le cose sono in parte migliorate dalla "stampa a percorsi ottimizzati", che evita l'inutile spostamento della testina quando la riga di stampa è composta di spazi agli estremi.



Nel modo di funzionamento C-64/128, selezionabile tramite il menu d'impostazione dei parametri di default (niente dip-switch), la MPS 1230 può essere predisposta per accettare solo il set di caratteri ASCII dei vecchi computer Commodore e i caratteri di controllo previsti da stampanti come la MPS 801, oppure riconoscere anche le sequenze escape previste dall'emulazione impostata. Le altre emulazioni disponibili sono la IBM Proprinter, la IBM Graphics Printer e la Epson FX 80, grazie alle quali è possibile adattare la stampante al tipo di software con il quale si desidera impiegarla.

Oltre al modo Draft, è previsto il modo NLQ, che a 10 cpi riduce la velocità di stampa nominale a 25 cps (il modo NLQ viene ottenuto con due passate sfalsate di pochissimo, quanto basta per stampare 18 punti verticali per carattere). La densità di caratteri per riga può variare fra 10 cpi e 30 cpi, così da ottenere da 80 a 240 caratteri per riga.

Oltre ai consueti stili di stampa, la MPS 1230 consente il "sopralineato", molto utile nel caso si debbano stampare testi tecnici nei quali si faccia uso di notazioni vettoriali. Sono anche disponibili la stampa proporzionale e i caratteri con larghezza doppia, mentre non è possibile la doppia altezza. Combinando insieme i modi grafici disponibili e i modi di stampa si possono selezionare circa 200 risultati diversi.

Accontentandosi di un buffer di stampa più piccolo (da 5K a 2,5K) è possibile caricare nella RAM della stampante un set di 94 caratteri ridefiniti, opzione molto utile per risolvere particolari esigenze di stampa senza entrare necessariamente in grafica.

Sono previste interlinee fisse e variabili, nonché lunghezze del foglio di stampa variabili fra 5 e 12 pollici. In grafica si riesce a raggiungere una densità di stampa orizzontale di 1920 punti.

La MPS 1230 dispone di un

programma interno per predisporre la configurazione di default tramite il quale è possibile variare numerosi parametri, come il tipo d'interfaccia, l'emulazione, il set di caratteri (con interfaccia parallela o seriale), il significato dell'indirizzo secondario in modo C-64/128, la presenza di un inseritore automatico di fogli singoli, la stampa mondirezionale (più precisa ma più lenta) e la stampa bidirezionale, il modo di stampa, la spaziatura, l'interlinea, lo zero sbarrato. Il menu viene attivato all'accensione e necessita del trascinamento a frizione per funzionare, in quanto ogni scelta provoca l'indietreggiamento della carta. L'interazione con il menu avviene semplicemente attraverso i tasti della console operativa.

Le impressioni d'uso

Abbiamo subito notato l'estrema lentezza che caratterizza il trascinamento del foglio quando si premono i tasti di avanzamento carta. Con ogni probabilità, questa stessa lentezza è la causa della non elevata velocità effettiva di stampa.

La qualità di stampa riscontrata è accettabile, anche se in modo Draft i caratteri appaiono poco definiti. Un punto a favore della MPS 1230 è sicuramente la bassa rumorosità, un particolare a cui talvolta non si dedica troppa attenzione.

L'inserimento dei moduli perforati è piuttosto scomodo per via del sistema di trascinamento a trattore che abbiamo già ampiamente criticato. Un uso continuo conferma poi l'assoluta mancanza di praticità nel passare da un tipo di trascinamento all'altro. Inoltre, la maggior parte delle stampanti appartenenti alla stessa fascia di prezzi ormai dispongono di un sistema di posizionamento automatico del foglio di stampa, per non parlare del parcheggio temporaneo del modulo continuo, assolutamente impensabile sulla MPS 1230.

Uno sguardo all'interno

La MPS 1230 è contraddistinta da una fragilità di fondo, denunciata soprattutto dalla sovrabbondanza di plastica e d'incastri. Per aprirla basta fare leva nei punti giusti, senza svitare nessuna vite.

All'interno, si nota che il carrello portatestina viene mosso tramite una cinghia dentata. Il nastro dell'inchiostro viene trascinato tramite un ingranaggio che è solidale con il carrello e ruota traslando lungo una cremagliera disposta parallelamente al rullo di stampa. Il carrello è vincolato a un unico albero che, contrariamente al solito, non consente di regolare la distanza fra la testina e la carta. Per quanto riguarda la meccanica di stampa, tutto sembra veramente ridotto all'osso.

Destano migliore impressione la logica di controllo e l'alimentazione. Niente sembra posticcio e l'assemblaggio è ordinato ed essenziale. La scheda di controllo è abbastanza compatta e sembra prodotta su larga scala. La EPROM con il firmware è montata su uno zoccolo per facilitarne la sostituzione. Considerando che sulla stessa scheda risiedono anche l'interfaccia parallela e quella seriale, si nota un alto livello d'integrazione e un basso impiego di componenti passivi.

L'alimentazione, infine, viene fornita da un compatto trasformatore racchiuso in un contenitore di plastica nero.

Conclusioni

Giunti al termine di questa prova possiamo affermare che se il prezzo della MPS 1230 fosse inferiore di un buon 30 per cento, i toni talvolta aspri con i quali abbiamo descritto alcune caratteristiche sarebbero stati molto più misurati; ma di fronte a un costo che con l'Iva supera il mezzo milione, non possiamo che sconsigliarne l'acquisto. Altre marche stanno dimostrando che si può fare molto di più a prezzi inferiori ed è questo il metro con il quale ci sembra doveroso giudicare un prodotto che nulla aggiunge al catalogo Commodore, se non un'anonima voce in fondo al listino prezzi. Ci auguriamo che in futuro vengano adottati modelli più vicini alle necessità, ma anche alle tasche, degli acquirenti a cui stampanti di questo tipo dovrebbero essere destinate. ■

DATI TECNICI

Testina di stampa:
9 aghi a impatto
Stampa bidirezionale con percorsi ottimizzati

Velocità di stampa:
120 cps Draft a 10 cpi
25 cps NLQ a 10 cpi

Densità di stampa:
10, 12, 15, 17,1, 20, 24, 30 cpi

Caratteri per linea:
Da 80 a 240

Trascinamento:
A trattore per moduli perforati, con trattore esterno a incastro.
A frizione per fogli singoli, funzionante solo a trattore non inserito

Emulazioni:
Epson FX 80
IBM Graphics Printer
IBM Proprinter

Interfacce di serie:
Seriale Commodore per C-64 e C-128
Parallela Centronics

Vita del caricatore d'inchiostro:
0,8 milioni di caratteri

Dimensioni:
94 x 370 x 253 mm

Peso:
4,2 Kg

Prezzo al pubblico:
L. 465.000 + Iva

Distribuzione:
Commodore Italiana spa
Viale Fulvio Testi, 280
20126 Milano
(Tel. 02/661231)

QUATTROMILA CARATTERI IN RGBI PER IL C-128

Indaghiamo sulle potenzialità del chip 8563: con l'interlace in modo testo e con un adatto software di gestione, è possibile disporre di 50 righe da 80 colonne

di Dario Greggio

Ci sono fondamentalmente due strade da seguire per realizzare un computer di successo. La prima è costruire una macchina dotata di un numero limitato di componenti, il cui buon funzionamento è garantito dalla casa produttrice o da anni di collaudi: in questo modo può nascere un calcolatore non eccessivamente sofisticato ma adatto a svolgere brillantemente impieghi di routine come la contabilità o l'archiviazione. La seconda strada è partire da un'idea innovativa, per la cui realizzazione si sono costruiti chip dedicati o comunque non esistenti prima (frutto magari di costosi investimenti nel settore della ricerca), al fine di realizzare un computer dalle capacità fuori dal comune e adatto a risolvere problemi per i quali il mercato non propone ancora una soluzione adeguata.

Si possono costruire macchine dell'uno o dell'altro tipo a qualsiasi livello: home computer, personal, workstation o mainframe. I risultati dipenderanno dalla situazione della tecnologia e del mercato, ma quasi sicuramente saranno positivi. L'Apple II e il PC XT della IBM sono esempi del primo tipo, mentre per il secondo potremmo citare il VIC-20, il C-64 e l'Amiga.

A volte però accade che qualche costruttore decida di assemblare su un circuito stampato tutti i componenti più collaudati che offre il mercato, anche senza dare all'insieme un'identità e una collocazione precisa. In pratica si ritiene che inserendo nella macchina tutto quello di cui si dispone ogni utente vi troverà qualcosa di utile... benché l'esperienza abbia dimostrato che computer del genere incontrano scarsissimo successo. Questo succede perché i tempi cambiano in fretta (e la tecnologia pure), ma soprattutto perché una macchina che offre prestazioni identiche a un'altra, aggiungendo tre o quattro caratteristiche proprie che però non risolvono nessuna esigenza particolare, difficilmente avrà un pubblico di ammiratori suoi. Al limite potrebbe interessare a chi intende studiarne l'hardware. In quest'ultima categoria sono da collocare computer come il Lisa della Apple e il Commodore 128.

Così si spiega perché un computer che contiene quasi 250K di memoria, 2 CPU, una marea di chip di contorno tra MMU, grafica e CIA, ha avuto un successo così scarso. Con questo articolo cercheremo di mettere in luce un'altra delle caratteristiche positive del C-128 che finora nessuno ha mai saputo (o voluto) utilizzare.

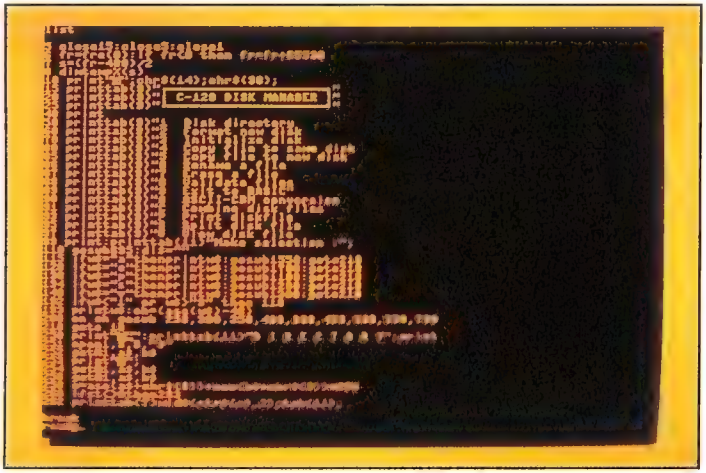
Si tratta dell'attivazione di un modo interlace analogo a quello dell'Amiga, al fine di raddoppiare la risoluzione verticale. Si riescono così a visualizzare 400 pixel in altezza senza alcuna aggiunta hardware e sul solito monitor RGBI della Commodore, il 1901. Poiché il sistema non riconosce questo modo grafico, è stato scritto anche un insieme di routine che vengono installate nel sistema in maniera trasparente e che gestiscono in modo testo il nuovo formato video.

Il chip 8563

Oltre al 6567 o VIC-II, conservato per mantenere la compatibilità con il C-64, il C-128 incorpora un secondo chip dedicato alla gestione video. Questo circuito integrato, l'8563, è molto più simile ai controller grafici dei PC IBM che al suo vicino di circuito stampato, in quanto adotta un tipo di gestione del tutto diverso da quello a cui ci ha abituato il C-64, sia per quanto riguarda la pagina grafica, sia per la memoria video e lo scambio di dati con la CPU. In pratica questo chip accede

esclusivamente a una memoria video dedicata, che quindi non appartiene a quella indirizzabile dalla CPU (nemmeno i suoi registri interni di controllo sono visibili sul bus indirizzi di sistema). Per comunicare con questo chip esistono due porte: agendo sulla prima si seleziona uno dei registri interni dell'8563, mentre nella seconda si memorizza o si legge il dato relativo al registro selezionato.

Tralasciamo comunque questi particolari che sono peraltro già noti agli utenti del C-128, per passare alla descrizione dei registri dell'8563 che ci riguardano più da vicino. I registri che vanno dal numero 0 al numero 8 vengono utilizzati per definire il tipo di segnale video (e quindi di modo grafico) inviato al monitor. È possibile programmare la posizione e l'ampiezza degli impulsi di sincronismo sia orizzontale sia verticale, il che si traduce in due importanti possibilità: quella di adattare il segnale video a qualunque monitor (standard o multisync, per esempio) e soprattutto quella di creare modi grafici personalizzati. Si giunge ad avere circa 900 pixel su una linea e 250 linee sullo schermo. Tra l'altro, anche le dimensioni della matrice che costituisce il carattere possono essere variate rispetto al default di 16 x 8 pixel. Si tenga presente che non è necessario che tutti i punti della matrice siano visualizzati.



Un listato Basic a 50 righe: si noti quanto chiaramente sono definiti i caratteri

Va tuttavia tenuto presente che parlare di aumento di risoluzione in questo caso è improprio, in quanto ciò che si ottiene non è la riduzione delle dimensioni dei pixel, ma la crescita dell'area dello schermo indirizzabile, a scapito del bordo. Si rischia quindi di avere un certo numero di pixel fuori schermo, a meno che non si disponga di un monitor speciale o comunque regolabile in altezza e larghezza. Esiste però un altro modo di aumentare la risoluzione in verticale, che consiste nell'attivare il modo interlace.

Anche se questa parola ricorda funesti sfarfallii a chi conosce l'Amiga, su un computer limitato come il C-128 può essere una scelta valida. Com'è noto, tramite questa tecnica si raddoppia il numero di linee visualizzate, mostrando sul video due immagini che si alternano sfalsate di mezza linea di schermo. Il risultato è una maggior risoluzione, legata però a un inevitabile sfarfallio dell'immagine dovuto al fatto che si scende al di sotto della soglia di persistenza delle immagini sulla retina.

Quando si attiva questo modo grafico (impostando i primi due bit del registro 8 del chip 8563), occorre anche alterare i valori dei registri relativi alla durata degli impulsi di sincronismo. Dato che sulle varie documentazioni consultate mancano precise indicazioni di questi valori, sono stati svolti alcuni calcoli di massima, correggendo poi i risultati sul banco di prova. Chi scrive si è inoltre scontrato con l'impossibilità di attivare contemporaneamente la grafica bitmap e l'interlace, pur avendo tentato diverse strade. Anche se non dovrebbero esserci limitazioni a livello teorico, il risultato è sempre l'apparizione di schermate in alta risoluzione incomplete o mal sovrapposte.

Un driver per il video

Quando si arriva a ottenere un'immagine in modo testo, in interlace, stabile e collocata correttamente nel raster video, il Commodore 128 funziona esattamente come prima, con l'unica differenza che i caratteri sono alti la metà e la parte di schermo gestibile è limitata alla metà alta. La parte inferiore dello schermo riporta invece caratteri corrispondenti al contenuto della pagina degli attributi video, che è contigua alla pagina dei caratteri. Questo accade perché il kernel del computer, non conoscendo il nuovo modo video, continua a fare riferimento alle locazioni di memoria per le quali è stato progettato, che non sono però adatte a gestire il nuovo schermo.

Ad agevolare la scrittura di un nuovo driver video viene la particolare struttura del kernel del C-128, che nonostante tragga le sue origini addirittura da quello del VIC-20 è in effetti un software piuttosto flessibile, anche se non è certo pari all'AmigaDOS o all'MS-DOS, e nemmeno a *GEOS*. Nei sistemi operativi più evoluti, infatti, esiste la tendenza a separare le routine che pilotano direttamente l'hardware da quelle che svolgono compiti di livello più elevato, e i due gruppi di

routine interagiscono attraverso opportune jump-table. Quando per esempio è necessario visualizzare un carattere, il software comunica tale esigenza alle routine di livello più basso, che provvedono a esaudire la richiesta.

Non c'è nessun motivo per cui le routine gerarchicamente superiori debbano avere qualche rapporto con l'hardware. Il vantaggio di questa impostazione è la possibilità di modificare l'hardware senza dover riscrivere completamente il software, ma adattandone solo la parte a più basso livello (ovvero il driver, le cui dimensioni sono in genere molto contenute). Per fare qualche esempio che tutti conoscono, si potrebbero citare i driver di stampa di *GEOS* o dell'Amiga, o i vari dispositivi (file di estensione ".SYS") del sistema MS-DOS.

Nel kernel del C-128 non esiste una struttura di driver, o meglio esiste ma è rigidamente memorizzata in ROM. Si può solo accedere a una tavola di salti in RAM (un po' più estesa di quella del C-64) attraverso la quale passano molte delle routine di I/O... molte, ma non tutte quelle che servono per riscrivere un driver. Per il nostro driver video siamo stati quindi costretti in alcuni casi a ricopiare parti del kernel non del tutto attinenti all'output dei caratteri, ma in compenso spesso si è fatto uso di alcune routine preesistenti alterandone semplicemente i parametri di input.

Con il modo video 80 × 50 è possibile utilizzare un solo colore per volta

La prima operazione è stata quella di assegnare una più ampia area di memoria al nuovo schermo, che è grande il doppio di quello precedente. Per fortuna la memoria video dedicata al chip 8563 si estende per 16K, e quindi vi si trova spazio sufficiente per raddoppiare le dimensioni della pagina testo. In precedenza, infatti, venivano utilizzati 8K per il set di caratteri e 2K rispettivamente per la pagina video e per quella degli attributi. In realtà tutto è diventato più semplice quando ci siamo accorti che il modo interlace presentava un'altra limitazione, ovvero l'impossibilità di definire il colore dei singoli caratteri: si può avere un solo colore per tutti i caratteri della pagina, da scegliere tra i 16 disponibili.

Quindi non ci sono problemi di memoria, 4K bastano anche con la nuova risoluzione. È sufficiente informare il chip video che gli attributi dei caratteri sono stati disabilitati e che la memoria schermo ora si estende per 4000 byte. Con questi valori vengono anche aggiornate alcune locazioni nella pagina zero del computer, locazioni che vengono utilizzate dalle routine del driver video.

A questo punto il compito si "riduce" a riscrivere o modificare tutte le routine del driver video preesistente al fine di realizzarne uno compatibile con il nuovo modo grafico. Tra queste routine troviamo quelle per l'output del carattere, quelle per lo spostamento del cursore, quelle per lo scroll del video, per la gestione dei modi insert e auto-insert e altre ancora. Una delle cose più odiose che si sono riscontrate durante l'analisi del vecchio driver e la scrittura del nuovo è stata la scoperta che il kernel, quando riceve dati dalla tastiera e deve generarne l'eco, non impiega la stessa routine di stampa a video che adotta quando l'esigenza è solo quella di generare un output a schermo.

Questo all'inizio ci ha lasciati sconcertati, in quanto il programma funzionava correttamente per l'output ma non altrettanto per l'input, che continuava a lavorare in zone di memoria sbagliate. Quando poi abbiamo compreso la ragione di questa anomalia, abbiamo constatato che le routine relative a questa sezione non facevano riferimento a nessun vettore in RAM, ma erano intercalate alle routine di più alto livello concernenti l'input da tastiera vero e proprio. Con una discreta dose di pazienza siamo comunque arrivati a riscrivere anche questa parte e il risultato finale ci sembra soddisfacente.

Considerazioni sui risultati

Il modo video che si ottiene con il nostro programma, definito 80 x 50, risulta per sua natura completamente trasparente al sistema. Questo significa che tutti i programmi che fanno un uso corretto delle risorse della macchina (cioè ricorrono alle sue routine di sistema come CHROUT, CHRIN e così via) potranno usufruire della

risoluzione video raddoppiata: tra questi s'inseriscono naturalmente i programmi Basic e il monitor. Programmi che eventualmente scavalcassero i vettori di collegamento del nostro driver, o peggio che vi si sovrapponessero in memoria sono destinati a bloccare il funzionamento della macchina.

È probabile che molti programmi professionali (i quali spesso prendono il controllo totale del computer) non funzioneranno. Se però la causa del malfunzionamento non risiede nell'accesso diretto al video, ma è solo un problema di aree di memoria, diventa possibile rilocare (con una certa pazienza) il driver del nuovo modo video affinché risieda in un'area che il pacchetto non utilizza.

Per quanto riguarda le caratteristiche intrinseche del driver che presentiamo, diciamo che la stampa di un carattere di controllo del colore di foreground modifica istantaneamente il colore di tutti i caratteri presenti sullo schermo. Il problema del passaggio da maiuscole a minuscole è stato risolto nella stessa maniera: il cambio del set di caratteri agisce su tutti i caratteri, come nel C-64. Inoltre è disabilitato il riconoscimento della combinazione di tasti Commodore-Shift, come anche la sequenza ESC-X che serviva a passare allo schermo a 40 colonne.

Vengono mantenute tutte le caratteristiche estese dell'editor di schermo del C-128, ossia le funzioni di Escape, lo scroll nelle due direzioni, i modi insert e auto-insert, i quattro tipi di cursore e così via. Il colore di fondo dello schermo può essere impostato con il tradizionale comando COLOR 6,<codice colore>: alla partenza del programma (o al suo restart ottenuto con SYS 4864) viene impostato per i caratteri il colore giallo su fondo azzurro, e viene abilitato il modo FAST.

Il driver è quindi perfettamente compatibile con i programmi che si servono delle routine di sistema, e non ci sono problemi a usare le istruzioni PRINT e LIST del Basic. Tuttavia istruzioni come WINDOW e CHAR fanno nascere una difficoltà. In pratica, l'istruzione WINDOW non accetta valori di riga superiori a 24, mentre l'istruzione CHAR, oltre a presentare il medesimo problema, non è utilizzabile in quanto accede al video direttamente. Questo è dovuto al filtraggio eseguito dall'interprete sui parametri numerici di queste istruzioni. WINDOW e CHAR possono essere comunque simulate con alcune istruzioni POKE o con brevi routine in LM; per quanto riguarda la definizione di finestre, un buon metodo alternativo può essere l'utilizzo delle sequenze ESC-T e ESC-B, come spiega il manuale del C-128. Le limitazioni citate sono purtroppo insuperabili in quanto per ovviarvi bisognerebbe riscrivere una grossa parte del Basic (il quale naturalmente non può fare uso di nessun driver in RAM per interpretare le istruzioni).

Oltre all'uso più banale, ossia una più semplice analisi dei programmi Basic o LM grazie alla maggior quantità di caratteri visualizzati, questo programma può servire come spunto per ulteriori esami del chip 8563, o come base per scrivere un word processor da 4 mila caratteri!

Listato: *80 x 50*

```
10 IFRGR(0)<>5 THEN PRINT"FUNZIONAMENTO IN MODO
   RGBI":END:ELSEFAST
20 FOR I=4864 TO 6790:READA:C=C+A:POKEI,A:NEXT
30 IF C<>227252 THENPRINT"ERRORE NEI DATA...":END
40 SYS4864
50 END
55 :
60 REM *** 80X50 IN RGBI ***
65 :
1000 DATA 169,0,141,0,255,162,36,189,32,19,32
1010 DATA 204,205,202,16,247,169,1,141,48,208
1020 DATA 169,11,141,17,208,76,80,19,234,234
1030 DATA 234,126,80,102,41,78,0,50,62,3,7,64
1040 DATA 7,0,0,0,0,0,0,0,0,8,0,120,8,32,7,210
1050 DATA 0,32,7,0,0,0,0,125,100,5,0,0,0,0,0
1060 DATA 0,0,0,0,0,0,169,240,141,38,3,169,19
1070 DATA 141,39,3,169,49,133,237,133,228,169
1080 DATA 99,141,36,3,169,23,141,37,3,169,147
1090 DATA 76,210,255,0,166,235,189,136,19,133
1100 DATA 224,133,226,189,186,19,133,225,9,16
1110 DATA 133,227,96,0,0,0,0,0,0,80,160,240
1120 DATA 64,144,224,48,128,208,32,112,192,16
1130 DATA 96,176,0,80,160,240,64,144,224,48
1140 DATA 128,208,32,112,192,16,96,176,0,80
1150 DATA 160,240,64,144,224,48,128,208,32,112
1160 DATA 192,16,96,176,0,80,0,0,0,0,1,1,1,2
1170 DATA 2,2,3,3,3,4,4,4,5,5,5,5,6,6,6,7,7
1180 DATA 7,8,8,8,9,9,9,10,10,10,10,11,11,11
1190 DATA 12,12,12,13,13,13,14,14,14,15,15,0
1200 DATA 76,8,21,72,165,154,201,3,240,3,76
1210 DATA 132,239,104,133,239,72,138,72,152
1220 DATA 72,173,33,10,208,251,133,214,169,195
1230 DATA 72,169,11,72,164,236,165,239,201,13
1240 DATA 240,38,201,141,240,34,166,240,224
1250 DATA 27,208,3,76,75,25,170,16,3,76,69,21
1260 DATA 201,32,144,191,201,96,144,3,41,223
1270 DATA 44,41,63,32,255,194,76,90,20,32,197
1280 DATA 20,232,32,133,203,164,230,132,236
1290 DATA 32,239,20,165,241,41,207,133,241,169
1300 DATA 0,133,245,133,243,133,244,96,166,243
1310 DATA 240,2,9,128,166,245,240,2,198,245
1320 DATA 36,246,16,9,72,32,225,24,162,0,134
1330 DATA 245,104,32,121,22,196,231,144,10,166
1340 DATA 235,228,228,144,4,36,248,48,22,32
1350 DATA 112,19,32,28,24,144,14,32,116,203
1360 DATA 176,8,56,36,248,112,4,32,155,20,24
1370 DATA 96,166,232,48,6,228,235,144,2,230
1380 DATA 232,166,228,32,114,19,164,230,228
1390 DATA 235,240,15,202,32,118,203,232,32,131
1400 DATA 203,202,32,248,22,76,167,20,32,215
1410 DATA 21,76,147,203,230,235,32,116,203,176
1420 DATA 249,198,235,32,112,19,164,231,132
1430 DATA 236,32,88,203,166,235,201,32,208,14
1440 DATA 196,230,208,5,32,116,203,144,5,32
1450 DATA 91,22,144,233,132,234,96,166,235,228
1460 DATA 228,144,14,36,248,16,6,165,229,133
1470 DATA 235,176,6,32,166,22,24,230,235,76
1480 DATA 112,19,201,27,240,56,166,245,208,8
1490 DATA 201,20,240,11,166,244,240,7,162,0
1500 DATA 134,239,76,94,20,162,13,221,140,199
1510 DATA 240,24,202,16,248,162,15,221,76,206
1520 DATA 240,4,202,16,248,96,76,96,26,240,29
1530 DATA 92,206,133,241,96,76,47,24,76,117
```

```
1540 DATA 200,96,41,127,201,32,144,9,201,127
1550 DATA 208,2,169,94,76,181,21,166,244,240
1560 DATA 5,9,64,76,94,20,201,20,208,3,76,225
1570 DATA 24,166,245,208,240,201,17,240,59,201
1580 DATA 29,240,209,201,14,240,33,201,18,208
1590 DATA 3,76,191,200,201,2,208,3,76,206,200
1600 DATA 201,15,208,3,76,220,200,201,19,208
1610 DATA 3,76,186,21,9,128,76,117,26,76,12
1620 DATA 206,0,96,32,239,20,32,116,203,176
1630 DATA 3,56,102,232,24,96,166,229,228,235
1640 DATA 176,249,32,93,200,198,235,76,112,19
1650 DATA 9,64,76,90,20,166,229,134,235,134
1660 DATA 232,164,230,132,236,132,233,32,114
1670 DATA 19,32,215,21,228,228,232,144,245,32
1680 DATA 80,193,76,112,19,164,230,32,133,203
1690 DATA 32,114,19,142,49,10,140,50,10,162
1700 DATA 24,32,218,205,41,127,32,204,205,162
1710 DATA 18,24,152,101,224,72,141,60,10,169
1720 DATA 0,101,225,141,61,10,32,204,205,232
1730 DATA 104,32,204,205,169,32,32,202,205,56
1740 DATA 165,231,237,50,10,72,240,20,170,56
1750 DATA 109,60,10,141,60,10,169,0,109,61,10
1760 DATA 141,61,10,138,32,62,197,162,18,24
1770 DATA 152,101,226,72,169,0,101,227,32,204
1780 DATA 205,232,104,32,204,205,173,61,10,41
1790 DATA 7,13,47,10,141,61,10,165,241,41,143
1800 DATA 32,202,205,104,240,3,32,62,197,174
1810 DATA 49,10,164,231,96,164,236,136,48,4
1820 DATA 196,230,176,15,164,229,196,235,176
1830 DATA 14,198,235,72,32,112,19,104,164,231
1840 DATA 132,236,196,231,24,96,166,241,44,166
1850 DATA 242,168,169,2,141,40,10,32,155,22
1860 DATA 152,164,236,72,32,230,205,104,76,202
1870 DATA 205,0,0,0,0,0,0,0,0,0,165,224,133
1880 DATA 226,165,225,9,16,133,227,96,166,229
1890 DATA 232,32,118,203,144,10,228,228,144
1900 DATA 246,166,229,232,32,133,203,198,235
1910 DATA 36,232,48,2,198,232,166,229,228,223
1920 DATA 176,2,198,223,32,220,22,166,229,32
1930 DATA 118,203,8,32,133,203,40,144,4,36,248
1940 DATA 48,203,96,32,114,19,164,230,228,228
1950 DATA 176,15,232,32,118,203,202,32,131,203
1960 DATA 232,32,248,22,76,220,22,76,245,25
1970 DATA 234,142,49,10,140,50,10,162,24,32
1980 DATA 218,205,9,128,32,204,205,32,230,205
1990 DATA 174,49,10,189,136,19,133,218,189,186
2000 DATA 19,13,46,10,133,219,162,32,24,152
2010 DATA 101,218,133,218,169,0,101,219,133
2020 DATA 219,32,204,205,232,165,218,32,204
2030 DATA 205,56,166,231,232,138,237,50,10,141
2040 DATA 50,10,162,30,32,204,205,162,32,165
2050 DATA 219,41,7,13,47,10,32,204,205,232,165
2060 DATA 218,32,204,205,32,249,205,173,50,10
2070 DATA 162,30,32,204,205,174,49,10,96,165
2080 DATA 153,240,3,76,21,239,165,236,133,233
2090 DATA 165,235,133,232,76,184,23,32,251,19
2100 DATA 32,33,25,165,208,5,209,240,250,32
2110 DATA 159,205,32,52,194,201,13,208,234,133
2120 DATA 214,169,0,133,244,32,197,20,142,48
2130 DATA 10,32,211,24,164,230,165,232,48,19
2140 DATA 197,235,144,15,164,233,205,48,10,208
2150 DATA 4,196,234,240,2,176,17,133,235,132
2160 DATA 236,76,217,23,152,72,138,72,165,214
2170 DATA 240,184,16,23,169,0,133,214,169,13
2180 DATA 162,3,228,153,240,4,228,154,240,3
2190 DATA 32,251,19,169,13,208,57,32,112,19
2200 DATA 32,88,203,133,239,41,63,6,239,36,239
2210 DATA 16,2,9,128,144,4,166,244,208,4,112
2220 DATA 2,9,64,32,255,194,164,235,204,48,10
2230 DATA 144,10,164,236,196,234,144,4,102,214
2240 DATA 48,3,32,28,24,201,222,208,2,169,255
2250 DATA 133,239,104,170,104,168,165,239,24
2260 DATA 96,72,164,236,196,231,144,7,32,239
2270 DATA 20,164,230,136,56,200,132,236,104
2280 DATA 96,138,10,170,189,60,24,72,189,59
2290 DATA 24,72,96,198,200,141,201,78,201,86
2300 DATA 24,165,200,171,200,127,26,212,200
2310 DATA 99,24,193,200,178,200,112,24,96,201
2320 DATA 201,24,165,236,72,32,197,20,32,239
2330 DATA 20,104,133,236,96,32,239,20,32,116
2340 DATA 203,176,3,56,102,232,24,96,32,181
2350 DATA 24,32,30,204,176,15,196,231,144,22
2360 DATA 166,235,232,32,118,203,176,14,32,192
2370 DATA 24,165,222,133,236,165,223,133,235
2380 DATA 76,112,19,32,28,24,32,88,203,32,91
2390 DATA 22,32,124,22,32,28,24,76,121,24,24
2400 DATA 96,166,229,228,235,176,249,32,103
2410 DATA 24,198,235,76,112,19,32,91,22,176
2420 DATA 236,208,233,230,235,208,237,165,241
2430 DATA 41,143,170,169,32,76,126,22,32,28
2440 DATA 24,176,1,96,76,103,24,32,116,203,144
2450 DATA 6,198,235,16,247,230,235,76,112,19
2460 DATA 32,30,204,32,197,20,228,223,208,2
2470 DATA 196,222,144,33,32,118,20,176,34,32
2480 DATA 91,22,32,88,203,32,28,24,32,124,22
2490 DATA 32,91,22,166,235,228,223,208,235,196
2500 DATA 222,208,231,32,192,24,230,245,208
2510 DATA 2,198,245,165,222,133,236,165,223
2520 DATA 133,235,76,112,19,36,215,16,38,32
2530 DATA 155,22,164,236,32,249,205,32,216,205
2540 DATA 141,51,10,41,240,133,219,32,249,205
2550 DATA 165,241,41,15,5,219,32,202,205,162
2560 DATA 10,173,43,10,76,204,205,201,27,208
2570 DATA 5,70,239,76,125,199,41,127,56,233
2580 DATA 64,201,27,176,10,10,170,189,108,25
2590 DATA 72,189,107,25,72,96,234,234,234,201
2600 DATA 25,236,202,21,202,233,202,18,26,10
2610 DATA 203,32,203,54,203,57,203,250,25,176
2620 DATA 203,81,203,225,202,228,202,71,203
2630 DATA 124,199,181,25,160,25,62,203,241,202
2640 DATA 19,202,253,202,57,26,71,26,85,205
2650 DATA 130,201,127,201,32,30,204,32,223,21
2660 DATA 230,235,32,112,19,164,230,32,116,203
2670 DATA 176,241,76,22,25,32,30,204,32,192
2680 DATA 24,196,230,208,5,32,116,203,144,238
2690 DATA 32,91,22,144,239,32,30,204,32,223
2700 DATA 21,230,235,32,112,19,164,230,32,116
2710 DATA 203,176,241,165,235,197,228,144,235
2720 DATA 240,233,76,22,25,32,30,204,138,72
2730 DATA 32,166,22,104,133,223,76,22,25,32
2740 DATA 215,21,76,247,195,32,155,20,32,86
2750 DATA 193,32,112,19,232,32,118,203,8,32
2760 DATA 129,203,40,176,3,56,102,232,96,32
2770 DATA 211,24,165,229,72,165,235,133,229
2780 DATA 165,248,72,169,128,133,248,32,184
2790 DATA 22,104,133,248,165,229,133,235,104
2800 DATA 133,229,56,102,232,32,86,193,76,112
2810 DATA 19,32,30,204,138,72,32,166,22,104
2820 DATA 133,223,76,22,25,32,30,204,32,116
2830 DATA 203,176,3,56,102,232,165,229,133,235
2840 DATA 32,155,20,32,133,203,76,22,25,189
2850 DATA 92,206,10,10,10,10,133,227,162,26
2860 DATA 32,218,205,41,15,5,233,76,204,205
2870 DATA 162,15,221,76,206,240,228,202,16,248
2880 DATA 96,169,0,160,216,76,16,206,0,0,0,0
```

CORSO DI PROGRAMMAZIONE IN C PER L'AMIGA: QUARTA PUNTATA

LA VERSIONE 5.0 DEL COMPILATORE C DELLA LATTICE

L'ultima versione del compilatore Lattice C introduce sostanziali miglioramenti. Dal sofisticato editor LSE a CodeProbe, il source level debugger che la Lattice include nel pacchetto

di Eugene P. Mortimore

Nel precedente articolo di avvicinamento al C abbiamo preso in considerazione le procedure d'impostazione dell'ambiente di programmazione: abbiamo esaminato un file script per la compilazione e il link di programmi in C che faceva uso del pacchetto Manx e dei relativi tool. In questo articolo prendiamo in esame un ambiente "antagonista", quello messoci a disposizione dalla recente versione 5.0 del compilatore Lattice C.

Durante i vari stadi evolutivi dell'Amiga la Lattice (come la Manx) ha prestato molta attenzione alle migliorie apportate all'Amiga, e ha sempre tenuto al passo i propri sistemi di programmazione. L'ultima versione, la 5.0, costituisce un pacchetto molto completo, adatto allo sviluppo di programmi semplici e di complesse applicazioni suddivise in più moduli. La versione 5.0 presenta le seguenti caratteristiche.

– Supporto per tutte le CPU Motorola 68000, 68010, 68020, 68030 e per il coprocessore matematico 68881.
– Un editor a più finestre, LSE (Lattice Screen Editor), integrato con il compilatore. Questa sistemazione fa sì che sia possibile vedere gli errori risultanti dalla prima compilazione e correggerli dall'editor, quindi rieseguire la compilazione, e continuare con altre eventuali correzioni. Ogni volta che si presenta un errore, l'editor colloca automaticamente il cursore nell'opportuno punto del sorgente. La Lattice fornisce due file ASCII, LSE.hlp e LSE.msg per personalizzare le operazioni di edit. Tutti i file sorgente vengono conservati in memoria per accelerare le fasi di edit e di compilazione.
– Un nuovo source level debugger full-screen anch'esso a finestre multiple, *CodeProbe* o cpr, che serve a sveltire le procedure di debug. Questo pacchetto consente di aprire più finestre di debug e di muoversi nei propri programmi istruzione dopo istruzione inserendo dove occorre dei breakpoint. CodeProbe ha un'interfaccia AREXX, la quale consente di controllarlo anche da altri programmi. Il disco contenente il sistema Lattice contiene anche circa 20 programmi AREXX d'esempio che mostrano come si può usare l'accoppiata AREXX-cpr.
– Un notevole "esaminatore" di programmi eseguibili, lprof, che effettua un'analisi delle prestazioni dei programmi mostrando dove, cioè in quali routine, viene sprecata la maggior parte del tempo macchina. Questo facilita l'ottimizzazione in termini di velocità.
– Un completo programma per la guida del processo di compilazione, lc, che permette di richiamare con un solo comando quattro tool del processo di compilazione, ossia le parti del compilatore (lc1 e lc2) corrispondenti alla prima e alla seconda passata, l'ottimizzatore (go) e il linker (blink).
– Una versione più evoluta del primo modulo del compilatore, lc1b, che genera automaticamente un file contenente i prototipi di tutte le funzioni incontrate nel modulo compilato.
– Un ottimizzatore del codice prodotto, go, che genera il codice in modo che sia più veloce o in alternativa più compatto.
– Un set completo di utility, alcune specifiche del Lattice e alcune di pubblico dominio, che forniscono utili strumenti per la costruzione di librerie e per personalizzare le procedure di programmazione.
– Librerie linked, di creazione Lattice, più veloci

e snelle, grazie alla codifica Assembly e ad algoritmi più efficienti.
– Osservanza del C ANSI standard.
– File INCLUDE precompilati, per accelerare la compilazione dei sorgenti.

A differenza del sistema Manx, che prevede la versione Developer e la versione Professional, del Lattice C esiste una sola versione. Il pacchetto è composto di cinque dischi, il primo dei quali contiene la maggior parte dei tool di programmazione come lc, lc1, lc2, l'editor LSE, i file di configurazione dell'editor e una serie di file utili alla personalizzazione del sistema. Il secondo disco contiene tutti i file relativi alle librerie linked dell'Amiga o peculiari del Lattice e la maggior parte dei file INCLUDE che servono alla programmazione. Il terzo disco contiene un certo numero di utility per la programmazione, come grep, cxref e lprof, la cui funzione è descritta nella Tavola 1. Il quarto disco contiene alcuni altri file INCLUDE in C e un certo numero di esempi che illustrano il modo d'impiegare il source level debugger CodeProbe. Il quinto disco contiene una serie di file INCLUDE Assembly che vanno usati insieme all'assembler asm della Lattice, che a differenza del sistema Manx costituisce un'applicazione a sé, necessaria soltanto se si programma in Assembly. Su questo disco si trova anche il file contenente gli ultimi aggiornamenti alla versione attuale, il file addendum.doc.

Tra gli altri file non descritti troviamo una serie di librerie di supporto matematico alla nuova release del software sistema, la 1.3, ossia routine per il calcolo di funzioni trascendenti per il coprocessore 68881 e nuove librerie in emulazione IEEE.

Nel pacchetto sono anche presenti un linker che gestisce gli overlay e diversi programmi di esempio, quasi tutti contenuti nel disco 4. La documentazione fornita consiste di circa 600 pagine suddivise in due raccoglitori ad anelli molto ben organizzati, sia per quanto riguarda le sezioni di riferimento sia per quelle tutorial. La documentazione è completa e ben illustrata, e fornisce semplici esempi che facilitano la comprensione. Il pacchetto può essere acquistato direttamente presso la Lattice.

Il prezzo complessivo è di 300 dollari ma vi sono diverse possibilità di risparmio se si possiede già una delle versioni precedenti del compilatore Lattice o del Manx; anche in Italia, inoltre, vi sono numerosi rivenditori di software che distribuiscono il sistema Lattice. Si tenga presente che analoghi sistemi Lattice sono disponibili in ambiente IBM o Atari e che molte delle opzioni qui discusse sono comuni a tutte le versioni di questo sistema, cosicché quanto si impara sull'Amiga ha valore anche in tutti gli altri ambienti.

Poiché il sistema Lattice nella sua completezza è molto vasto e dispone di un gran numero di opzioni, in questo articolo non le esamineremo tutte. Concentreremo invece la nostra attenzione sull'organizzazione generale del sistema e sul significato pratico della maggior parte dei file che s'incontrano durante l'uso del pacchetto.

I dispositivi logici nel sistema Lattice

Il sistema Lattice definisce il suo ambiente di lavoro attraverso l'assegnazione di una serie di dispositivi logici all'interno dei quali i tool si aspettano di trovare i file di cui hanno bisogno. Occorre dunque che attraverso il comando ASSIGN vengano associati ai dispositivi logici previsti dai tool i corretti cammini di ricerca (path) nelle directory dei vari dischi. Le assegnazioni dei dispositivi logici sono le seguenti.

– INCLUDE: è il nome del dispositivo logico al quale viene associata la directory che contiene i file INCLUDE in C. Questa indicazione viene scavalcata dall'opzione –i del compilatore.
– LC: viene associata alla directory in cui si trovano i tool del compilatore come lc, lc1 o lc2, e altri programmi eseguibili.
– LIB: specifica dove il tool lc troverà le librerie linked. Questo tool usa tale definizione per creare il suo file lnk.
– QUAD: indica la posizione in cui la prima passata del compilatore dovrà scrivere il file temporaneo (file quad) da sottoporre a lc2, la seconda parte del compilatore.

Il pacchetto della Manx, invece, non impiega i dispositivi logici per l'individuazione dei file, e adotta alcune variabili globali impostabili tramite il comando SET. Il vantaggio è che a una singola variabile possono essere associati diversi cammini di ricerca che vengono sondati sequenzialmente, mentre un dispositivo logico può essere associato sempre a un solo cammino di ricerca. In questo senso il sistema Manx è senz'altro più flessibile.

Grazie al source level debugger CodeProbe, da poco distribuito con il pacchetto, sono ora disponibili per gli utenti Lattice capacità di debug pari se non superiori a quelle offerte dal debugger SDB, utilizzabile con il pacchetto Manx. Questo potente strumento, che consente un'analisi riga per riga, funzione per funzione, semplifica enormemente il processo di programmazione in C.

I tool di programmazione

La Tavola 1 nella pagina successiva elenca i più importanti file eseguibili del sistema Lattice: la descrizione che li accompagna dovrebbe essere sufficiente a renderne chiara la funzione. Nella tavola i file vengono suddivisi in due categorie: i primi nove, LSE, lc, lc1, lc2, blink, cpr, go, lc1b e asm, sono rispettivamente l'editor; il tool che permette di automatizzare compilazione, ottimizzazione e fase di link; le due passate del compilatore; il linker; il source level debugger; l'ottimizzatore; la versione evoluta della prima passata del

Nome del tool	Posizione	Descrizione
Tool di programmazione		
LSE	Disco 1	Editor di programmi sorgente con più finestre. Agevola la compilazione in memoria
lc	Disco 1	Tool per richiamare il compilatore, l'ottimizzatore e il linker con un solo comando
lc1	Disco 1	Prima passata del compilatore
lc2	Disco 1	Seconda passata del compilatore
blink	Disco 1	Linker a gestione di overlay
cpr	Disco 3	Source level debugger per C e Assembly
go	Disco 3	Tool per ottimizzare i codici prodotti
lc1b	Disco 3	Versione con prototipi della prima passata del compilatore
asm	Disco 5	Assembler. Non viene coinvolto nella compilazione di programmi in C
Utility		
build	Disco 3	Utility per inserire linee di testo in un file in una data posizione
cxref	Disco 3	Utility per generare cross-reference dei file sorgente in C
diff	Disco 3	Utility per confrontare i file sorgente carattere per carattere
dumpobj	Disco 3	Utility per effettuare il dump dei file oggetto
extract	Disco 3	Utility per stampare i nomi dei file presenti in certe directory
fd2pragma	Disco 3	Utility per creare un file "pragma". In unione alla direttiva #pragma (che dev'essere utilizzata per indicare tutte le funzioni di sistema che si usano nel sorgente), questo file consente di ottimizzare le chiamate alle routine di sistema dell'Amiga, facendo in modo che i loro argomenti siano passati direttamente nei registri anziché sfruttando lo stack, come previsto dal C
files	Disco 3	Utility per cercare, copiare o cancellare file e directory
grep	Disco 3	Utility per la ricerca delle espressioni ricorrenti di caratteri nei testi
lcompact	Disco 3	Utility per compattare i file INCLUDE e quindi accelerare la compilazione
lmk	Disco 3	Utility per gestire la trasformazione in eseguibile di un programma composto da diversi moduli sorgente
lprof	Disco 3	Utility per analizzare la ripartizione del tempo della CPU e individuare eventuali inefficienze
lstat	Disco 3	Utility per gestire e confrontare i dati forniti da lprof
patch	Disco 3	Utility per prelevare gli aggiornamenti al sistema Lattice provenienti da un BBS
omd	Disco 3	Utility per disassemblare i moduli oggetto, generando codice misto C e Assembly
oml	Disco 3	Utility per la creazione e la gestione delle librerie linked
splat	Disco 3	Utility per la sostituzione delle stringhe nei file comandi
tb	Disco 3	Utility per la gestione delle exception (usa un file SnapShot.tb)
touch	Disco 3	Utility per aggiornare la data e l'ora di un file
wc	Disco 3	Utility per contare il numero di caratteri, parole e righe contenute in un file

Tavola 1: elenco riassuntivo dei tool di programmazione e delle utility della versione 5.0 del pacchetto di compilazione Lattice

compilatore e l'assembler. La seconda categoria riguarda le utility del sistema Lattice; tutti questi file eseguibili si trovano nel disco 3.

Analogamente al pacchetto Manx, il Lattice è un sistema di programmazione molto complesso. Comprende cinque tool separati ma interdipendenti, i tool di programmazione, che a un primo esame possono risultare di difficile comprensione nonostante la linearità dei manuali. Per questa ragione, conviene esaminare i significati dei file previsti da ogni tool durante la fase di scrittura e di compilazione dei programmi.

Per ora, facciamo l'ipotesi di non utilizzare il tool lc: a questo scopo si esamini la Figura 1 nella pagina seguente, nella quale per semplicità si considera la compilazione di un programma in C composto di un solo file sorgente. Nella figura, i rettangoli rappresentano i vari tool di programmazione. Iniziando da sinistra, troviamo l'editor LSE, il compilatore suddiviso nelle due passate lc1 e lc2, il linker blink e infine il debugger cpr. Come abbiamo già detto, l'assembler asm non viene utilizzato in questa catena di programmazione ma costituisce uno strumento a sé stante.

È buona norma trasferire tutti i file eseguibili dal disco distribuito con il pacchetto alla directory c del disco sistema, quello che risiede nel disk drive df0:. È importante ricordare che l'editor e il source level debugger non sono indispensabili in questa catena, mentre lo sono il compilatore a due passate e il linker. L'ordine di esecuzione dei programmi va da sinistra a destra, come indicano le frecce della figura. Si noti che alcuni dei file sono prodotti automaticamente dai tool di sistema, mentre altri sono controllati dal programmatore mediante opzioni da specificare agli stessi tool per modificarne il funzionamento.

Nella figura appaiono molti file di input e di output. Tranne il file ff prodotto dal linker, tutti gli altri non sono eseguibili. I file di input sono quelli che si trovano al di sopra dei rettangoli raffiguranti i tool mentre i file di output sono quelli che si trovano sotto. I file di input, anche quelli che non sono in formato ASCII, sono i file letti e utilizzati dai vari tool; i file di output sono file creati dai tool. Si noti che spesso i file prodotti da un tool diventano l'input dei tool successivi. Per sapere quali di questi file sono testi ASCII esaminabili dall'utente, si veda la Tavola 2 (a pagina 100).

Si ricordi sempre che tutti i tool, specialmente quando vengono usati singolarmente, producono anche un certo output di testo sul video: questo output, tuttavia, può essere ridiretto in un file con l'apposito comando AmigaDOS (si indica il nome

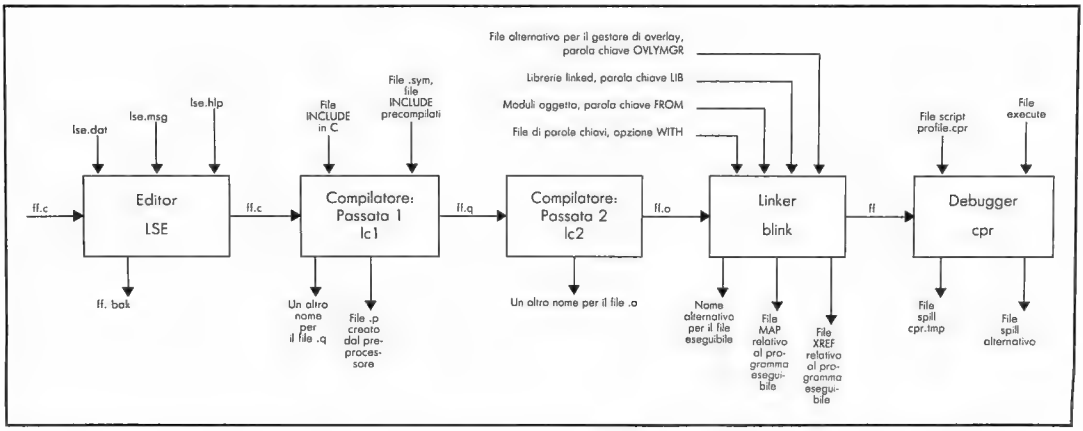


Figura 1: i tool del sistema Lattice nella sequenza con cui vengono impiegati. Si notino i file di input e di output che intervengono

di un file preceduto da un segno di maggiore). I file così prodotti non sono ovviamente mostrati nella Figura 1.

L'editor di testi LSE

Il primo rettangolo partendo da sinistra rappresenta l'editor di testi LSE; di solito, vi sono al massimo quattro file di input e due di output legati a questo tool. Il primo è il file sorgente in C, qui chiamato ff.c: si tratta del testo sorgente che costituisce il programma da compilare. D'ora in avanti considereremo sempre questo file come esempio di file sorgente. Naturalmente può accadere che il programma sia formato da più file sorgenti, e in tal caso il processo di edit illustrato nella figura si applicherà a ognuno di questi file: ossia ciascun file verrà letto nei buffer interni dell'editor, verrà modificato dall'utente e in seguito salvato nuovamente su disco con lo stesso nome.

Un altro file richiesto dall'editor è lse.dat, che contiene la più recente configurazione dell'editor salvata dall'utente. Questo file consente la personalizzazione di molte operazioni dell'editor e viene generato attraverso l'uso del programma LSEinst (installazione dell'LSE).

Gli altri due file di input sono lse.msg (il file dei messaggi) e lse.hlp (il file di help), i quali contengono rispettivamente le definizioni dei messaggi e degli help visualizzati sullo schermo durante l'edit dei file. Essendo file di testo, possono essere modificati dall'utente. Il disco 1 del pacchetto Lattice fornisce alcuni esempi sia per questi due ultimi file sia per lse.dat. I due file di output dell'editor sono il nuovo file e la sua copia di backup, un file con lo stesso nome ma con l'estensione .bak, che contiene la versione precedente del file.

Il tool lc1

La prima passata del compilatore sottopone il sorgente al preprocessore, e ne svolge un'analisi sintattico-lessicale per individuare possibili errori. Prevede tre file di input e tre file di output e, come nel caso dell'editor, alcuni sono obbligatori e altri no. Vengono riconosciuti anche i file INCLUDE del pacchetto Manx, che possono essere considerati il quarto tipo di possibili file di input.

Il primo file di input è ff.c, il file sorgente da compilare. Il secondo comprende tutti i file INCLUDE in linguaggio C che il programma richiama al suo interno tramite la direttiva #include (ognuna di queste direttive provoca l'inclusione del file indicato all'interno del sorgente nel punto in cui si trova la direttiva). La directory in cui si trovano questi file dev'essere quella indicata dal dispositivo logico INCLUDE:, ma se l'utente desidera che vengano impiegati i file INCLUDE che si trovano in un'altra directory può scavalcare il dispositivo logico attivando l'opzione –i quando chiama il compilatore, e fornendo un nuovo cammino di ricerca. Si possono quindi distinguere i file INCLUDE in primari (quelli che risiedono nella directory logica INCLUDE:) e secondari (quelli che vengono indicati di volta in volta al compilatore). Si tratta di una suddivisione di comodo che viene fatta dal programmatore in base alle specifiche informazioni richieste dal programma.

I file INCLUDE precompilati rappresentano il terzo tipo di file di input di lc1. Il compilatore accetta infatti sia file INCLUDE normali, quelli in formato ASCII, sia file precompilati. Questi file, che si riconoscono per il suffisso .sym, sono di solito più brevi degli altri e permettono dunque di risparmiare spazio sul disco e tempo in fase di caricamento; il loro impiego è dunque tanto più consigliabile quanto più è esteso il lavoro. Per costringere il compilatore a leggere questi file occorre specificare l'opzione –H.

Per il compilatore Lattice i file INCLUDE precompilati costituiscono un'importante novità rispetto alle precedenti versioni, che lo porta allo stesso livello del Manx e rende obsoleti i precedenti file INCLUDE compressi. La discussione del-

l'opzione –p fornirà altre informazioni su questi file.

Lc1 produce in output al massimo tre file. Il primo è un file .q temporaneo, di tipo quad, che in seguito verrà letto dal tool lc2: questo file viene di solito salvato nel RAM disk, e viene automaticamente cancellato dopo essere stato letto da lc2. Il secondo file di output è semplicemente lo stesso file quad con un altro nome, che viene generato se si attiva l'opzione –o del compilatore.

Il terzo file di output (la tavola dei simboli) viene prodotto solo se si ricorre all'opzione –p seguita dall'indicazione del nome con cui dev'essere salvato il file e dal nome della directory in cui verrà inserito. Oltre che servire per l'analisi e lo studio delle variabili usate, l'opzione –p consente anche di generare tavole di simboli relative ai file INCLUDE, cioè di traformarli in file INCLUDE precompilati (ovviamente, si trasformano solo quelli che si prevede resteranno invariati).

Nella discussione dei file di input del tool lc1 si è parlato dell'opzione –H e di come i file precompilati contribuiscano ad accelerare il processo di compilazione. Si noti che –p sottopone il sorgente solo al preprocessore e non al compilatore, cioè fa in modo che non venga generato il file temporaneo .q e non venga svolta alcuna analisi sintattico-lessicale. Se invece si desidera sia la compilazione, sia la generazione del file .sym, occorre attivare l'opzione +H.

Infine, se si prevede d'impiegare il debugger CodeProbe, occorre fin da questa fase specificare l'opzione –d# (dove "#" indica un numero da 0 a 5), la quale arricchisce il file quad delle informazioni necessarie al funzionamento del debugger CodeProbe; si veda a questo scopo la descrizione del tool lc.

Il tool lc2

Questo tool richiede un solo file di input, il file temporaneo quad, e genera un solo file di output, il modulo oggetto vero e proprio. Il file temporaneo, nel nostro esempio ff.q, è prodotto dal tool lc1 e dopo essere stato utilizzato dal tool lc2 viene automaticamente cancellato.

Il modulo oggetto che viene generato ha lo stesso nome del file sorgente ma con estensione .o, quindi nel nostro esempio si chiamerà ff.o. Se si desidera alterarne il nome, è possibile utilizzare l'opzione –o seguita dal nuovo nome.

Occorre notare che il tool lc2 genera il modulo oggetto senza chiamare l'assembler, a differenza di quanto avviene con il pacchetto Aztec C68k realizzato dalla Manx Software Systems Inc., nel quale l'assembler è parte integrante e indispensabile di qualsiasi compilazione. Per via di questa particolare struttura del compilatore, con il sistema Lattice non abbiamo la possibilità d'inserire linee in Assembly all'interno dei sorgenti in C, e neanche di esaminare la traduzione in Assembly svolta dal compilatore. Comunque, il Lattice contiene il tool omd per disassemblare i moduli oggetto.

Il linker blink

Il linker blink accetta un massimo di cinque file di input. Il primo è il modulo oggetto, il file ff.o prodotto nella seconda fase della compilazione e memorizzato sul disco nella stessa directory in cui si trova il file sorgente ff.c. Questo file viene creato dal compilatore.

Il secondo è un file che contiene un insieme di opzioni che altrimenti dovrebbero essere digitate sulla linea comando: il linker cerca questo file solo se viene invocato specificando la parola chiave WITH. Questa semplificazione nel modo di fornire le opzioni al linker è solo una possibile alternativa al metodo tradizionale: se si utilizza il tool lc oppure un opportuno file comandi di compilazione e link, le opzioni vengono indicate una volta per tutte in questo file e non serve utilizzare la parola chiave WITH.

I rimanenti file di input del linker sono gli altri moduli oggetto relativi al programma, ossia i moduli oggetto derivanti dalla compilazione delle altre sezioni di codice sorgente. Questi file sono associati alle parole chiave ROOT e FROM. Per esempio, se il nostro programma consiste di due moduli sorgente, f1.c e f2.c, dapprima si provvede a compilarli separatamente producendo tre moduli oggetto (rispettivamente f1.o e f2.o), quindi si effettua il link indicando, dopo la parola chiave FROM, come file primario il modulo f1.o seguito dai nomi degli altri due file oggetto.

Analogamente, si possono indicare al linker delle librerie in formato oggetto indicandone il nome subito dopo l'opzione LIB: per esempio, due librerie linked che andranno sempre chiamate in causa saranno la libreria standard Lattice e la libreria contenente le funzioni dell'Amiga, rispettivamente chiamate lc.lib e amiga.lib. Si ricordi che le librerie linked contengono i corpi di funzioni impiegate ma non definite all'interno dei programmi sorgente, come per esempio le funzioni printf e scanf: si tratta di funzioni per lo più codificate in Assembly che vengono unite al programma principale nella fase di link.

L'ultimo file di input del linker è il cosiddetto file di gestione degli overlay. Perché venga preso in considerazione dal linker occorre indicare la parola chiave OVLYMGR (serve soltanto durante la creazione di grandi programmi).

Il tool blink può produrre un massimo di tre file di output. Il primo è il programma eseguibile, il risultato finale della compilazione. Se non viene indicato diversamente attraverso la parola chiave TO, il linker assegna a questo file lo stesso nome del file sorgente privato dell'estensione .c: nel nostro esempio di Figura 1 si chiamerà quindi ff.

Gli altri file di output del linker sono opzionali e sono controllati dal programmatore, che può deciderne la creazione attraverso particolari opzioni. Si può generare un file cross-reference attraverso la parola chiave XREF, cioè un file contenente un elenco di tutti i simboli utilizzati nel programma, dei nomi delle variabili, delle locazio-

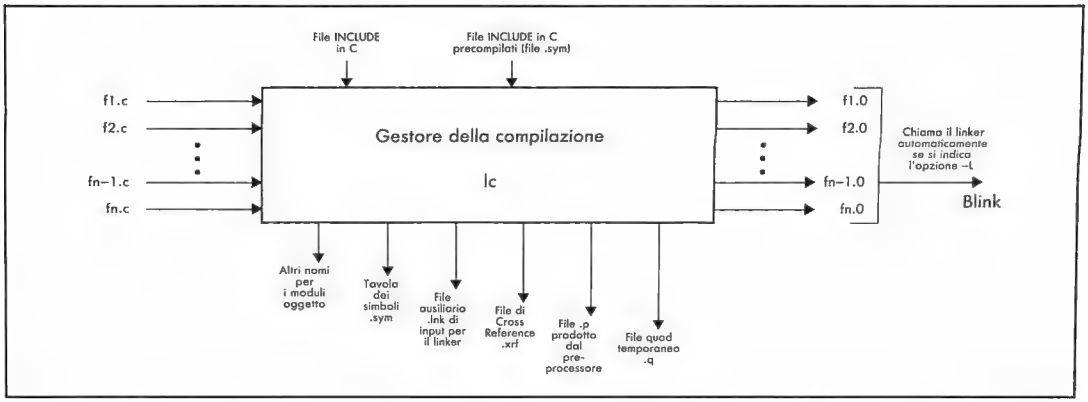


Figura 2: i file di input e di output del tool lc, che automatizza la compilazione, l'ottimizzazione e la fase di link

ni a cui è assegnato un nome, e di altre informazioni utili per vedere dove sono allocati i vari oggetti di cui fa uso il programma. Il terzo file, che viene generato se s'indica la parola chiave MAP, mostra la dislocazione delle varie parti del programma negli hunk.

Il source level debugger CodeProbe

Il rettangolo più a destra nella Figura 1 rappresenta l'ultimo tool della catena di programmazione del sistema Lattice, il debugger CodeProbe. Questo tool, il cui nome è abbreviato in cpr, accetta al massimo tre file in input.

Il primo è il programma eseguibile, nel nostro esempio ff, prodotto in output dal linker blink. I due file successivi sono opzionali. Il primo è un file comandi per il debugger, chiamato profile.cpr. Il debugger lo manda automaticamente in esecuzione quando viene invocato: serve per definire la configurazione generica di questo tool e per impartire automaticamente una serie di comandi predefiniti. Si possono avere diversi file comandi, differenziandoli a seconda dei programmi che stiamo scrivendo. Ciascun file profile.cpr può essere creato in modo da adattare il debugger al particolare programma in esame e per fargli eseguire la serie di comandi più adatta alla particolare situazione.

Si possono anche creare alcuni file comandi addizionali per il debugger, i cosidetti file execute, che vengono mandati in esecuzione attraverso il comando Execute del debugger. Analogamente al caso precedente, in questi file possono essere inserite le sequenze di comandi che vengono utilizzate più spesso, oltre a quelle eseguite all'avviamento del debugger.

Il tool cpr produce un solo file di output. Si tratta del file spill, che viene creato automaticamente per accedere il più velocemente possibile alle informazioni di debug. Per accelerare le cose tale file viene posto nel RAM disk con il nome cpr.tmp, ma la posizione e il nome possono essere alterati utilizzando l'opzione –temp del debugger.

L'uso dell'assembler Lattice

Si sarà notato che nella Figura 1 non compare l'assembler asm: questo perché, contrariamente a quanto accade nel sistema Manx, qui l'assembler può essere usato solo separatamente. I programmi scritti in Assembly si possono compilare utilizzando asm, e successivamente si possono unire agli altri moduli oggetto (anche a quelli originati da sorgenti in C) tramite il linker.

L'assembler può avere al massimo due file in input. Il primo è il file sorgente in Assembly, che chiamiamo ff.asm, mentre il secondo rappresenta i file INCLUDE in Assembly.

In output, il tool asm produce al massimo due file. Il primo è il modulo oggetto, che salvo scelte esplicite (tramite l'opzione –o) viene salvato con lo stesso nome del sorgente, ma con l'estensione .asm. Il secondo file di output è il file list in Assembly, che viene prodotto se è stata specificata l'opzione –l. Questo file riporta tutte le istruzioni Assembly del file sorgente e il loro equivalente in linguaggio macchina; sono anche indicati gli indirizzi relativi (offset rispetto alla prima istruzione) delle singole istruzioni del programma. In realtà non viene creato un file vero e proprio: il suo contenuto viene inviato allo schermo come output del programma. Di conseguenza, per intrappolarlo all'interno di un file occorre che l'output venga ridiretto tramite l'AmigaDOS.

Il tool lc

Uno degli strumenti più comodi messi a disposizione dal pacchetto Lattice è il tool lc. I file che lo riguardano sono elencati nella Tavola 2, mentre la Figura 2 mostra le sue interazioni con gli altri tool del sistema.

Lo scopo di lc è fornire una strada semplice per unificare sotto un unico comando le chiamate ai tre tool necessarie per trasformare in eseguibile uno o più file sorgente. Facendo uso di lc, con un solo comando si possono compilare, ottimizzare e sottoporre a link sia singoli file sorgente sia

Nome del file	Posizione e opzioni relative	Descrizione del contenuto
File di input e di output del programma d'installazione dell'editor LSE, LSEinst		
LSE.dat	Specificata dall'utente	File di dati per l'editor riguardante la configurazione Amiga (Non ASCII)
File di input dell'editor LSE		
ff.c	Specificata dall'utente	File sorgente in C: al massimo nove file in due finestre (ASCII)
LSE.dat	Specificata dall'utente	File di dati dell'editor (ASCII)
LSE.msg	Specificata dall'utente	File contenente i messaggi dell'editor (ASCII)
LSE.hlp	Specificata dall'utente	File contenente i dati di tutti gli schermi di help (ASCII)
File di output dell'editor LSE		
ff.c	La stessa di partenza	File sorgente modificato (ASCII)
File di input del tool lc		
ffn.c (n=1..N)	La stessa di ff.c	File sorgente in versione definitiva (ASCII)
file INCLUDE	INCLUDE:	File INCLUDE che per default sono nel dispositivo logico INCLUDE: (ASCII)
file INCLUDE	Specificata dall'utente (–i)	File INCLUDE relativi al programma che non risiedono nel dispositivo logico INCLUDE: (ASCII)
file .sym	Specificata dall'utente (–H)	File INCLUDE precompilati o tavole dei simboli (Non ASCII)
File di output del tool lc		
ff.o	Specificata dall'utente	File oggetto prodotto dal compilatore (Non ASCII)
nome-qualsiasi	Specificata dall'utente (–o)	Nome e directory alternativa per il file oggetto (Non ASCII)
ff.sym	Specificata dall'utente (+H)	File INCLUDE precompilato o tavola dei simboli (Non ASCII)
ff.lnk	Specificata dall'utente (–L)	File prodotto dal compilatore per istruire il linker (ASCII)
nome-qualsiasi	Specificata dall'utente (–g)	Cross-reference e file list prodotti dal compilatore lc1b (ASCII)
ff.q	Specificata dall'utente (–q)	File temporaneo di passaggio da lc1 a lc2 (Non ASCII)
ff.p	Specificata dall'utente (–p)	File di output del preprocessore, usato in seguito per la compilazione (ASCII)
file .lib	Specificata dall'utente (–R)	Libreria linked da aggiornare con i moduli oggetto appena creati (Non ASCII)
File di input della prima passata del compilatore (lc1 e lc1b)		
ff.c	Specificata dall'utente	Uno dei file sorgenti in versione definitiva (ASCII)
file INCLUDE	INCLUDE:	File INCLUDE che per default sono nel dispositivo logico INCLUDE: (ASCII)
file INCLUDE	Specificata dall'utente (–i)	File INCLUDE relativi al programma che non risiedono nel dispositivo logico INCLUDE: (ASCII)
file .sym	Specificata dall'utente (–H)	File INCLUDE precompilati (Non ASCII)
File di output di lc1 e lc1b		
ff.q	Specificata dall'utente	File temporaneo di passaggio dalla prima alla seconda passata (Non ASCII)
nome-qualsiasi	Specificata dall'utente (–o)	Nome alternativo per il file temporaneo (Non ASCII)
ff.p	Specificata dall'utente (–p)	File di output del preprocessore (ASCII)
nome-qualsiasi	Specificata dall'utente (–g)	Cross-reference prodotto dal tool lc1b (ASCII)
File di input della seconda passata del compilatore (lc2)		
ff.q	Specificata dall'utente	File quad temporaneo (Non ASCII)
File di output della seconda passata del compilatore (lc2)		
ff.o	Stessa del sorgente	File oggetto (Non ASCII)
nome-qualsiasi	Specificata dall'utente (–o)	Nome e directory alternativa per il file precedente (Non ASCII)
ff.sym	Specificata dall'utente (+H)	Tavola dei simboli (Non ASCII)
File di input dell'assembler asm		
ff.a	Specificata dall'utente	File sorgente in Assembly (ASCII)
file INCLUDE	Specificata dall'utente (–i)	File INCLUDE in Assembly (ASCII)
File di output dell'assembler asm		
ff.o	La stessa di ff.a	Modulo oggetto (Non ASCII)
nome-qualsiasi.o	Specificata dall'utente (–o)	Nome alternativo per il modulo oggetto (Non ASCII)
ff.lst	La stessa di ff.a (–l)	Listato in LM del file compilato. Appare a video, ma può essere ridiretto (ASCII)
File di input del linker blink		
ff.o	La stessa del file ff.a	Modulo oggetto (Non ASCII)
nome-qualsiasi	Specificata dall'utente (WITH)	File contenente le opzioni per il linker (ASCII)
file oggetto	Specificata dall'utente (FROM)	Altri file oggetto da sottoporre a link (Non ASCII)
nome-qualsiasi	Specificata dall'utente (LIB)	Librerie linked (Non ASCII)
nome-qualsiasi	Specificata dall'utente (OVLYMGR)	File da usare al posto di quello standard per il controllo degli overlay (Non ASCII)
File di output del linker		
ff	La stessa di ff.c (*)	Programma eseguibile (Binario)
nome-qualsiasi	Specificata dall'utente (TO)	Nome alternativo per il programma eseguibile (Binario)
nome-qualsiasi	Specificata dall'utente (XREF)	Cross-reference dei simboli del programma (Non ASCII)
nome-qualsiasi	Specificata dall'utente (MAP)	File MAP che mostra il contenuto degli hunk che costituiscono il programma, oltre ai simboli, le librerie, ecc. (ASCII)
File di input del debugger cpr		
ff	La stessa di ff.c	Programma eseguibile (Binario)
nome-qualsiasi	Specificata dall'utente	Nome alternativo per il programma eseguibile (Binario)
profile.cpr	Specificata dall'utente	File comandi di startup per il debugger (ASCII)
nome-qualsiasi	Specificata dall'utente	File execute del debugger (file comandi) (ASCII)
nome-qualsiasi	Specificata dall'utente	Nome aggiuntivo per il file eseguibile, usato nel debug in multitasking (Binario)
File di output del debugger		
cpr.tmp	Specificata dall'utente	File spill del CodeProbe, di solito posto nel RAM disk (Non ASCII)
nome-qualsiasi	Specificata dall'utente (–temp)	File spill alternativo (Non ASCII)

Per il tool lc: le opzioni –e e –j controllano quali errori vengono riportati nel file ridiretto
Per i compilatori lc1 e lc1b: l'opzione –j controlla quali errori vengono riportati a video

(*) Occorre specificare l'opzione ADDSYN del linker se si desidera che il file eseguibile contenga le informazioni di debug per il CodeProbe in uno dei suoi hunk

Tavola 2: sommario di tutti i file di input e di output previsti dai tool di programmazione del sistema Lattice versione 5.0

programmi complessi.

Se si ha un programma composto da più file sorgente da modificare, lc può gestire la situazione con facilità. Per esempio, se il nostro programma è composto dai tre moduli sorgente ff0.c, ff1.c e ff2.c, il programma eseguibile si ottiene direttamente con la semplice linea comando sotto riportata, senza effettuare altre operazioni.

```
LC -L ff0.c ff1.c ff2.c
```

Il tool lc può inoltre essere usato per aggiungere automaticamente altri moduli oggetto a una libreria linked, richiamando in maniera automatica l'utility di gestione delle librerie oml, quando c'è bisogno di aggiornare con una versione più recente un modulo oggetto precedentemente incluso in una libreria.

Come si vede nella Tavola 2, tutti i file con estensione .c che vengono presentati all'ingresso di lc causano la chiamata del compilatore ed eventualmente del linker allo scopo di produrre un modulo eseguibile. Per esempio, la Figura 2 mostra N file (f1.c..fN.c) che vengono indicati come file di input per il tool lc; se si specifica l'opzione –O, i file quad generati per ogni sorgente vengono ottimizzati.

Inoltre, il tool lc può ricevere in input uno o più file INCLUDE in formato ASCII e uno o più file INCLUDE precompilati. Come con la prima passata del compilatore, lc1, per quelli ASCII occorre indicare la locazione tramite l'opzione –i (a meno che non si trovino nella directory logica INCLUDE:), mentre per quelli precompilati occorre indicare il nome dopo l'opzione –H.

Questo tool può produrre una lunga serie di file di output. In primo luogo genera un file oggetto per ogni file sorgente in C sottopostogli in input: nella Figura 2 questi file si trovano sulla destra. Inoltre, il tool lc può produrre fino a sei file ausiliari. Il primo, in basso a sinistra, è semplicemente il file oggetto prodotto in precedenza, il cui nome è stato cambiato mediante l'opzione –o, la stessa prevista da lc2. Il secondo file ausiliario è la tavola dei simboli, un file con l'estensione .sym, che viene prodotto attivando l'opzione +H, la stessa prevista da lc1. Il terzo file prodotto è il file ausiliario per il linker, un file con l'estensione .lnk, all'interno del quale il compilatore inserisce tutte le informazioni di cui il linker ha bisogno per trasformare i file sorgenti in eseguibili. Questo file viene generato attivando l'opzione –L, la quale si occupa anche d'invocare il linker. Si può infatti fare in modo che il tool lc invochi automaticamente il linker blink se non si sono verificati errori durante la compilazione; blink utilizza poi le librerie linked standard, lc.lib e amiga.lib per risolvere i riferimenti esterni. Esistono 14 opzioni del tool lc per controllare il linker dalla linea di comando, e tutte sono ampiamente commentate nella documentazione della Lattice.

Il quarto file di output è il file cross-reference, utile per studiare come il programma è strutturato in memoria. L'opzione che sovrintende la scrittura di questo file è –g, ma ne esistono altre nove che permettono un controllo più preciso dei dati forniti dalla cross-reference.

Il quinto file generato è quello contenente l'output del preprocessore, un file con l'estensione .p, che viene prodotto se si specifica l'opzione –p, la stessa prevista da lc1; anche in questo caso esistono alcune sotto-opzioni per controllare il file generato. L'ultimo file è il solito file quad, di estensione .q.

Tra le sue tante opzioni, il tool lc ne contiene alcune che servono a indicare se e come si desidera che il compilatore generi nel file oggetto le informazioni necessarie perché il linker inserisca in un hunk del programma eseguibile (e non in un file a parte come con il Source Level Debugger della Manx) i dati necessari al funzionamento del source level debugger CodeProbe. Per scegliere quanti dati devono essere inclusi nel file eseguibile, si usano le sei opzioni –d0...–d5. Con –d0, per esempio, non vengono accluse informazioni per il debugger, mentre con –d5 vengono generate informazioni relative a tutti i simboli e le strutture dichiarate nel programma, anche quando non vengono utilizzati. I livelli intermedi consentono la scelta di quantità più limitate d'informazioni.

Lc permette anche un dettagliato controllo degli errori di compilazione: se si specifica l'opzione –C, lc procede alla compilazione degli altri file sorgente anche quando uno ha dato origine a errori fatali; in questo modo è possibile portare avanti la compilazione e tornare sugli errori solo alla fine.

Un'altra opzione interessante è –M, che ordina a lc di compilare solo quei file sorgente la cui data di creazione è più recente della data dei rispettivi file oggetto. Se per esempio il nostro programma è composto da sei file sorgente e ne modifichiamo uno, rimandando in esecuzione lc assisteremo alla compilazione del solo file modificato, e poi al link di tutti i moduli.

Il prossimo articolo

Nel prossimo articolo esamineremo in dettaglio alcuni file INCLUDE. Impareremo così a orientarci nella complessità delle strutture predefinite del software sistema dell'Amiga, in particolare dell'Exec e di Intuition.

Vedremo anche qual è la filosofia adottata per assegnare i nomi delle strutture di dati che popolano il software sistema dell'Amiga, confrontandola con i metodi usati nel C tradizionale. Studieremo come vengono definiti i flag di Intuition e come si possono combinare i bit relativi a diversi flag per ottenere ciò che desideriamo. ■

Per ulteriori informazioni contattare direttamente:

Lattice Incorporated
2500 South Highland Avenue
Lombard, IL 60148, USA
(Tel. 001/312/9161600 – Fax 001/312/9161190)

ARRETRATI

COMPLETATE LA VOSTRA COLLEZIONE

NUMERO 1/86 - Telecomunicazioni: guida all'acquisto di un modem. Il disk drive 1541 ed i suoi limiti tecnici. Corso di programmazione in L.M. su C-64. Il C-128 prova. Il nuovo Commodore Amiga a confronto con IBM PC, AT e Macintosh. Reset per il 1541. Amiga, dove fantasia e realtà si incontrano. Fare musica con il C-128. Convertitore grafico per Commodore 64: listato. Cultura e informatica. Software Gallery: Lode Runner Rescue, The Hitchhiker's Guide to the Galaxy, Jet, Scenery Disk. Software Helpline: The Hitchhiker's Guide to the Galaxy, Zaxxon, Wolfenstein, Star Wars, Ghostbusters, Jumpman, Mindshadow e Tracer Sanction.

NUMERO 2/86 - Jack Tramiel. Corso di programmazione in L.M. su C-64. Telecomunicazioni. Il C-128D. Come operare all'interno del disk drive. Computer e grafica. La Commodore e la didattica. La Rom del C-64. Amiga e Atari 520ST a confronto. Speciale USA: il CES di Las Vegas, il mondo del III Commodore Show di San Francisco. La mappa di memoria del C-128. Fare musica con il 64: Rondò Veneziano. Archivio programmi: listato per C-64. Software Gallery: Turbo Loading Cartridge, Machine Lightning, Basic Lightning, Uridium, Software Helpline: The Hitchhiker's Guide to the Galaxy, Beyond Castle Wolfenstein, Ghostbusters, Cosmic Balance, Jumpman, Star Trek, Avventura nel computer.

NUMERO 3/86 - Nuovi prodotti Commodore: 64C, 1581 e 1802. Linea Commodore PC IBM compatibili. Gli Ampersand file. Geos. Corso di programmazione in L.M. su C-64. Il SIM HI-FI IVES. Software in CP/M per C-128. Amiga: una nuova era nel mondo del PC. La A-Squared ■ AmigaLive! Le piante parlano: progetto hardware/software. Amiga: tra sogno e realtà. Espansioni di memoria per il C-128. I disk drive 1570 e 1571 a confronto. Prom ed Eprom. Il suono nel C-128. Jenny: listato per C-128. Eseguiamo le routine del 1541. Suoni e musica con l'AmigaBASIC. Software Gallery: Di-Sector V3.0, Project: Space Station, Matrix 128, 3D Graphics Drawing Board, Textcraft, Game Killer, Nexus, Mach 128, 1571 Clone Machine, VizaStar e VizaWrite, The Final Cartridge. Software Helpline: The Dallas Quest, Questron.

NUMERO 4/86 - Smau '86. CLI: Command Line Interface. La compatibilità IBM per Amiga. Sidecar e Transformer. Schema interno dell'Amiga. Genlock: effetti speciali audio e video con l'Amiga. Amiga days '86. Geos e il suo creatore: intervista. Personalizziamo il sistema operativo del C-64. Csa turbo Amiga. La memoria di massa e i compact disk. Terzo Commodore Show, Los Angeles. Tempo di Biennale, tempo di Amiga. Protezione scrittura ■ interruttore per il numero di device: progetto hardware per C-64. Software per Amiga: descrizione di più di 250 programmi. Grafica 128: listato per C-128. Corso di programmazione in L.M. su C-64. Lista delle routine Kernel del C-128. Software Gallery: Johnny Reb II, War Play, Time Trax, Aegis Impact, The Cataloger, Cartuccia ES-9, Online!, Deluxe Paint. Software Helpline: Borrowed Time.

NUMERO 1/87 - Il C-64 a Hollywood. L'Amiga nella realizzazione del serial «Amazing Stories». Amiga e Pontaccio. La politica economica della Commodore. Gli Idea Processor. World of Commodore servizio speciale. Amiga 1060 Sidecar. La grafica, l'Amiga ■ l'AmigaBASIC. Grafica avanzata per il C-128: programmazione del chip 8563. Geos: le chiavi del regno. Super Basket NBA: simulazione per C-64/128. Commodore PC 40 AT. Hardcopy della pagina grafica 640x200 pixel per C-128. Gestione magazzino per C-128. Routine alternate Kernel-Disco. La stampante MPS 1000. Corso di programmazione in L.M. su C-64. Software Gallery: Tomahawk, Flight Simulator II, Partner 128, Deluxe Paint II, Defender of the crown, Blood'n guts, '43 one year after, Mind Walker, Thai boxing, Skyfox. Software Helpline: Flight Simulator II, View to a kill, Ultima IV.

NUMERO 2/87 - Il Consumer Electronics Show: servizio speciale. Nuovi nati nella famiglia Commodore: Amiga 500 e 2000. Nuove istruzioni del chip 8502 utilizzato come CPU nel Commodore 128. La videodigitalizzazione: due noti digitalizzatori per C-64/128 e Amiga. Batch file con l'Amiga. Corso di programmazione in L.M. per C-64. La produzione di «The Bard's Tale»: intervista in esclusiva. Gestione di un club: listato per C-64. Protector 128: protettore di programmi per C-128. Geos: le chiavi del regno. Nuove potenzialità per le macchine ■ 8 bit della Commodore. Copie bit a bit sul 1541. Software Gallery: American Challenge e America's Cup, Page Setter, The Inheritance, Webstars, Vera Cruz, Agent Orange, S.D.I., Sinbad, Turbo Pascal, EDNA, Power Cartridge. Software Helpline: Oo-Topos.

NUMERO 3/87 - Commodore Italiana: gli assi nella manica per il 1987. I nuovi Commodore Amiga 500 e 2000. CeBit '87: servizio speciale. Roma Ufficio '87. La famiglia di programmi Geos: le applicazioni dedicate e la nuova versione per C-128. Corso di Programmazione in L.M. su C-64. Find 128: utility per C-128. Commodore Show San Francisco: servizio speciale sulla nota rassegna californiana. Il Software Sistema dell'Amiga. Catalogazione Dischi per C-128: listato per C-128. Il disk drive OC-118: prova hardware. Dump 128: listato. Velocizzatori e disk drive 1541. Scroll list 128: listato. Software Gallery: Balance of Power, Silent Service, Portal, Graphic Adventure Creator, Melody Hall's Printware Series, Cyborg, Twin Tornado, Thanatos, The Ket Trilogy. Software Helpline: Leather Goddesses of Phobos.

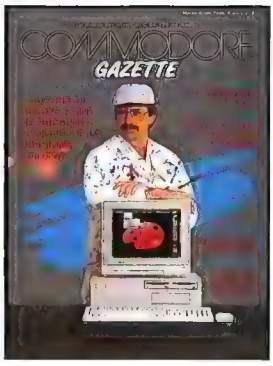
NUMERO 4/87 - Obiettivo Telematica: la telematica e il C-64. La Commodore cambia volto: servizio speciale. Basic 8.0: sistema grafico per Commodore 128. Desktop publishing con l'Amiga: rassegna di programmi di desktop publishing realizzati per l'Amiga. La stampante MPS 1200: prova hardware. Amigraf: listato in Amiga BASIC per tracciare il grafico di funzioni matematiche. Il Consumer Electronics Show di Chicago: Servizio speciale. Nuovi comandi e Ram Disk: Utility per C-128. Disk Sector Editor: listato per C-128. L'hardware dell'Amiga. Bank Data: archivio di nomi e indirizzi per C-128. Corso di Programmazione in LM per C-64. Software Gallery: Bureaucracy, Faery Tale, Uninvited, Murder on the Atlantic, Deathscape, Frankenstein, Up Periscope!, Barbarian, 500 cc Grand Prix, Logistix, Superbase. Software Helpline: A View to a Kill.

NUMERO 5/87 - Computergrafica nella ricerca scientifica: l'uso dell'Amiga. Software musicale per l'Amiga. L'Amiga 2000 e la Compatibilità IBM: un'introduzione al modello 2000 e alla A2088. Inventando l'Amiga: il travagliato avvio commerciale del gioiello della Commodore. Corso di Programmazione in L.M. per C-64. L'Hardware dell'Amiga: le caratteristiche strutturali dell'Amiga. Il Comdex di Atlanta: servizio speciale. Codici in data C-128: listato. Amiga 3D: animazione in 3D con l'Amiga. Append Merge: listato per C-128. Semper Sperum: listato per C-64 per la pianificazione delle giocate al Lotto. Bank Data: La produzione di Adventure nel nostro Paese. Software Gallery: the Mirror Hacker package, ProWrite, Guild of Thieves, Barbarian, Defender of the Crown, Denarius, Revs Plus, Inheritance 2, Kinetik, The Three Musketeers, Galileo. Software Helpline: Trinity.

NUMERO 6/87 - SIM Hi-Fi. SMAU. Dossier speciale: i pirati in Italia. La storia del gruppo 2703; un nuovo business: l'hardware pirata; un pirata pentito; la riscossa degli importatori; quando la copia costa più dell'originale; il periodo dell'impunità è finito. Prove hardware: il disk drive da 3,5" 1581. Telematica: il Videotel per il C-64. Amiga: le nuove frontiere dei giochi d'avventura. Toolkit 128: listato per nuovi comandi BASIC. Corso di programmazione in LM per C-64: decima puntata: il sistema BCD. Amiga hardware: gli sprite e il blitter. I nuovi prodotti USA per Amiga. Elab C-128: listato di un programma grafico. Software Gallery: ACE 2, Pirates!, Passengers on the wind, Lurking horror, Hollywood hi-jinx, Vampyr Motel, I predatori della Valle dei Re, Vizawrite desktop, 007: The living daylights, Solomon's key. Software Helpline: The lurking horror, Bureaucracy.

NUMERO 7/87 - Le promesse del 1988: intervista ai manager della Commodore. Novità per Amiga dagli USA: AmiEXPO, Commodore Show e Comdex. Anteprima sugli emulatori di C-64 per Amiga. L'Aids del computer: i programmi virus. Fred Fish: i programmi di pubblico dominio per Amiga. La geometria frattale: le teorie di Mandelbrot e un programma in Amiga Basic. Le espansioni di memoria 1764, 1700 e 1750 per C-64/128. Prova del PC1. Videoscape 3D: il desktop video. C-128: l'avvenire si chiama Geos. Amiga hardware: ultima puntata. Corso di programmazione in LM per C-64: assemblatori e codice sorgente. Software Gallery: Advanced OCP art studio, Alien fires, Echelon, Bubble bobble, Knight orc, Quedex, Fire power, Test drive, Video title shop, The hunt for Red October. Software Helpline: The three musketeers.

NUMERO 1/88 - Novità dagli Usa per l'Amiga. AmiEXPO e World of Commodore Show. Intervista con Dale Luck sui nuovi Kickstart e Workbench 1.3. Speciale telematica: il 6499 alla prova; Amiga ■ Videotel, intervista con Hugo Cornwall: la telematica in 60 parole chiave. Wb e info file: ■ cosa servono e come usufruirne. Un viaggio nell'universo dei giochi di ruolo. Il quinto convegno Automat. Geos: anatomia di un sistema operativo. Corso di programmazione in LM per C64. Equo canone C-64: listato. Gli astri attraverso il C-128: listato. Software Gallery: Airborne ranger, Apollo 18: mission to the moon, Chuck Yeager's advanced flight trainer, Auto duel, Moebius, Shoot'em up construction kit, The art of chess, World tour golf, Bocce, Tombola, Flying shark, Mini putt, Crazy car, Test drive. Software Helpline: Neverending story, Stationfall.



NUMERO 2/88 - Intervista con l'autore di Deluxe Paint. Speciale simulazione del volo: Flight Simulator II; Il funzionamento di Flight Simulator II; Co-Pilot un libro da volare; il volo a schema; simulatori di volo a confronto. Reportage dal CEBIT '88. Sconfitta l'AIDS dell'Amiga, con il listato del programma antivirus "Guardian". Prova hardware: hard disk per l'Amiga, geoPublish, il desktop publisher GEOS compatibile. Corso di programmazione in LM per il C-64. Pangea Basic C-128, listato. Super Quark C-128, listato. Il progetto ARP. Software Gallery: Shadowgate, Stratton, Yogi Bear, Terramex, Ports of Call, The Graphics Studio, Jet, John Brenner Boston Celtic, Power at Sea, Stealth Fighter, Card Sharks, Mercenary, Strike Fleet, King of Chicago, Jinxter. Software Helpline: Perry Mason and the Case of the Mandarin Murderer.

NUMERO 3/88 - I programmatori del mondo Amiga a confronto. I PC Commodore IBM compatibili della terza serie. Una rete telematica sul mondo. Come crearsi un videogioco "fatto in casa". Come far publicare un programma. La programmazione in ambiente Geos con Geoprogrammer. Suddividiamo in moduli i nostri programmi Basic. Geos: anatomia di un sistema operativo. Un archivio elettronico per il vostro Amiga. I drive ottici: trenta hard disk da 20 MB in un CD Rom. Corso di programmazione in LM per C-64. Software Gallery: Shakespeare, Video Titler, Corporation, Ferrari Formula One, Geos 117, Return to Atlantis, Rimrunner, Intellitype, Skyfox II: The Cygnus Conflict, Crack, Io, Advanced Tactical Fighter, Tetris, Troll. Software Helpline: John Brenner, Boston Celtic, Beyond Zork, Curiosità e informazioni utili.

NUMERO 4/88 - Videoscape 2.0 e le nuove frontiere dell'immagine. L'Amiga, le immagini, la Polaroid Palette. Gestire i testi in ambiente Geos: Geowrite Workshop. La MPS 1500 C: i colori economici. Il servizio Videotex nel mondo. Il più grande software del mondo. CES: i nuovi videogiochi alla mostra di Chicago. Geos: anatomia dei menu e delle icone. Un C-128, un telescopio e una stella cometa. Corso di programmazione in L.M. per il C-64. A caccia di immagini nella memoria del C-64. Software Gallery: Lasertem, Vixen, Express Paint, The three Stooges, Patton vs Rommel, BeckerBasic, Personal Newsletter, Enlightenment Druid II, Publisher Plus, Sentinel, F/A 18 Interceptor, Capone, The Armageddon Man, Questron II. Software Helpline: Labyrinth, Shadowgate, Curiosità e informazioni utili.

NUMERO 5/88 - ...E la Sublogic creò il volo. Matematica a tre dimensioni per l'Amiga. Okimate 20, una stampante dai colori brillanti. Un database per il mondo di Geos. Un tavolo da disegno per l'Amiga. La gestione multitasking, i tipi di codice e gli interrupt. Chicago: l'hardware e il software all'Amiexpo '88. Geos: anatomia di un sistema multitasking. Corso di programmazione in LM per il C-64. Software Gallery: Bard's Tale III, Carrier Command, Breach, Halls of Montezuma, Wasteland, PIXmate, Legend of the Sword, Whirligig, Corruption, The President is Missing. Software Helpline: Che cos'è un'avventura?, Curiosità e informazioni utili.

NUMERO 6/88 - Schede acceleratrici per un Amiga "turbo". Le stampanti Mannesmann MT 87 e MT 88. Novità Commodore: non certo troppe. Una visita al mondo Commodore. Le stampanti Fujitsu DX2300 e DL3300. Un foglio elettronico per il mondo di Geos. Grafica avanzata con le espansioni RAM 1700 e 1750. Corso di programmazione in C per l'Amiga. Come programmare i gadget stringa di Intuition. La struttura ExecBase dell'Amiga. Software Gallery: Ultima IV, Ultima V, Starfleet I, Bobo, Game Over II, Caveman Ugh-lympics, Bubble Ghost, Barbarian II, Starglider 2, Red Storm Rising, Fusion, Virus, Rocket Ranger, Typhoon. Software Helpline: Sherlock Holmes.

NUMERO 1/89 - L'Intelligenza Artificiale sull'Amiga e il pacchetto Magellan. Le stampanti OKI a 9 aghi e a 18 aghi. La parola ai programmatori: sta nascendo l'Amiga 3000? Geos: come costruire un box di dialogo. Un C-128 a caccia di eclissi. Corso di programmazione in C per l'Amiga. Come programmare i gadget booleani di Intuition. ExecBase: i parametri dinamici e le liste di sistema. Le macro istruzioni Assembly dell'Amiga. Software Gallery: Battle Chess, Microprose Soccer, The Universal Military Simulator, Who framed Roger Rabbit, Neuromancer, Deathlord, Driller, Falcon, Times of Lore, Captain Blood, Elite, The Mars Saga, Total Eclipse, Simulgolf. Software Helpline: Ultima V, Pool of Radiance.

NUMERO 2/89 - L'hard disk A590: 20 MB più espansione RAM per l'Amiga 500. 68020 e sistema operativo Unix per l'Amiga 2000. La violenza corre sul video. CEBIT '89: tutte le novità della Commodore. Corso di programmazione in C per l'Amiga. Cygnused professional, un editor per i programmatori. I gadget proporzionali di Intuition. Arexx e Wshell, due utility per il programmatore. L'Amiga Basic alla conquista di Intuition: schermi e finestre. Frattali in Interlace con 320x200 pixel a quattro colori. Software Gallery: War in the Middle Earth, Police Quest, Wec Le Mans, Blasteroids, The legend of Blacksilver, Empire, Italy '90 Soccer, Pac-Land, Draw 2000, TV Sports Football, R-Type & Denaris, Zany Golf, Purple Saturn Day, Last Duel. Software Helpline: Zak McKracken and the Alien Mindbenders.

NOME .....

COGNOME .....

INDIRIZZO .....

CITTÀ .....

C.A.P. .....

DESIDERO RICEVERE I SEGUENTI NUMERI ARRETRATI:

.....

PER UN TOTALE DI LIRE (lit. 16.000 ognuno):

.....

☐ **ALLEGO ASSEGNO BANCARIO O CIRCOLARE**

☐ **ALLEGO FOTOCOPIA DELLA RICEVUTA DEL VAGLIA POSTALE**

FIRMA.....

RITAGLIARE E SPEDIRE A: **COMMODORE GAZETTE**
Via Monte Napoleone, 9 - 20121 Milano

L'ESTENSIONE EXTEND DELL'AMIGA BASIC

Il pacchetto Extend aggiunge all'Amiga BASIC quarantasette comandi per gestire finestre, gadget, menu, memoria, schermi, requester, e per comunicare con l'AmigaDOS

di Eugene P. Mortimore

Programmando in Amiga BASIC ci si scontra ben presto con i suoi limiti nelle comunicazioni con Intuition, e quindi con la difficoltà di creare interfacce user-friendly che siano all'altezza delle possibilità offerte dall'Amiga. Sebbene una delle sue doti migliori sia da sempre la possibilità di gestire gli schermi, le finestre e i menu di Intuition attraverso istruzioni semplici (le consuete SCREEN, WINDOW, MENU, MENU ON, MENU OFF, MENU STOP, ON MENU e altre), presenta però alcune difficoltà nell'uso dei sotto-menu, dei gadget e dei requester. La Microsoft non ha mai cercato di risolvere questi problemi.

Per esempio, anche se la libreria Intuition è predisposta per farlo, l'Amiga BASIC non consente di aggiungere, rimuovere, abilitare o disabilitare i gadget, di esaminarne lo stato in seguito a interventi da parte dell'utente, di variarne gli attributi video come il colore e il modo grafico.

Per quanto riguarda i menu, il discorso è molto simile. Nonostante Intuition lo consenta, l'Amiga BASIC non permette di disporre a piacere le varie voci dei menu, di assegnare hot key o di gestire le scelte effettuate dall'utente nei vari livelli di sotto-menu.

Non è inoltre possibile modificare i colori del testo di un requester, creare requester stringa o gestire un requester per scegliere i file presenti su un disco. Non si può effettuare il refresh della finestra di output del BASIC, né cambiare i titoli delle finestre, usare fonti e stili personalizzati al loro interno, muovere gli schermi verticalmente, salvare una schermata in formato IFF, caricare e visualizzare immagini in formato IFF.

Infine, l'interprete non prevede alcun metodo per eseguire comandi del DOS, o per ricavare informazioni sui file, costringendo i programmatori a escogitare trucchi di ogni tipo, come chiamare la funzione Execute dell'AmigaDOS.

Qualcuno potrebbe puntualizzare che tramite opportune chiamate dirette alle funzioni di sistema si riesce a sopperire a gran parte dei limiti enunciati, come dimostra per esempio l'articolo "L'Amiga BASIC alla conquista di Intuition: schermi e finestre" pubblicato nel numero scorso. È vero, ma per riuscirci è necessario approfondire la conoscenza del sistema ben oltre la semplicità del Basic, uno sforzo che non tutti i programmatori in Basic sono disposti a fare. Insomma, su una macchina come l'Amiga chiunque si aspetterebbe un Basic più evoluto.

Tutte le operazioni che i comandi dell'Amiga BASIC non prevedono, e altre ancora, sono ora possibili con il Basic esteso distribuito dalla Sunsmile Software: il programma Extend. Tutte le istruzioni fornite da questo pacchetto si trovano in una libreria di nome extend.library, a cui si può accedere semplicemente copiando la libreria nella directory logica LIBS:, e il relativo file extend.bmap, nella directory all'interno della quale teniamo i file .bmap.

Le funzioni di Extend

La tavola proposta nella pagina successiva elenca le funzioni fornite da Extend, ovvero le 47 nuove funzioni presenti nella extend.library (ogni comando è seguito da una breve descrizione che illustra il suo funzionamento). Le categorie in cui si possono raggruppare i nuovi comandi sono otto: nella prima troviamo quelli relativi alla gestione delle finestre, che permettono di manipolare le finestre di output dell'Amiga BASIC in un modo molto simile a quello disponibile quando si pro-

LE ISTRUZIONI BASIC FORNITE DA EXTEND

Nome della funzione	Descrizione
Istruzioni per la gestione delle finestre	
CHANGE	Cambia il titolo di una finestra
REFRESH	Cerca di effettuare il refresh della finestra di output dell'Amiga BASIC
SETWINDOW	Comunica alla libreria dell'Extend informazioni sulla finestra del Basic
Istruzioni per la gestione dei gadget	
CHECK	Controlla se uno dei gadget indicati come argomenti è stato selezionato dall'utente
CLEARBUFFER	Cancella il contenuto del buffer di un gadget stringa
GADBCOLOR	Imposta gli attributi video dei bordi di tutti i gadget
GETSTRING	Restituisce il contenuto di un gadget stringa in una variabile Basic
OFFGAD	Disabilita un particolare gadget
ONGAD	Abilita un gadget
PUTSTRING	Inserisce nel buffer di un gadget stringa il contenuto di una variabile Basic
REMOVEGAD	Rimuove un dato gadget dalla finestra del Basic
SETHILIGHT	Imposta gli attributi di evidenziazione dei gadget
SREQUEST	Imposta gli attributi di un gadget stringa variabile
STATUS	Restituisce il numero del gadget stringa nel quale si trova il cursore
TOGGLEOFF	Porta un gadget toggle allo stato OFF
TOGGLEON	Porta un gadget toggle allo stato ON
WAITBOOLGAD	Attende che un gadget booleano venga selezionato dall'utente
WAITGAD	Mette l'Amiga BASIC in attesa finché un certo gadget non diventa attivo
Istruzioni per la gestione dei menu	
CHECKSUB	Controlla se un particolare sotto-menu è stato aperto dall'utente
COMMANDKEY	Assegna una sequenza di tasti a una voce dei menu
MAT	Modifica gli attributi di evidenziazione di una voce dei menu
SETSUB	Imposta la posizione di un sotto-menu rispetto al menu padre
SUBCOLOR	Imposta il colore di tutti i sotto-menu di un menu
SUBMENU	Collega un sotto-menu alla voce di un altro menu
Istruzioni per la gestione della memoria	
CLOSELIB	Libera la memoria precedentemente allocata per i gadget
FREEMEM	Libera tutta la memoria allocata con il comando MEMORY
MEMORY	Alloca un certo spazio di memoria
Istruzioni per la gestione degli schermi	
MSCREEN	Muove verticalmente lo schermo
XSCROLL	Effettua lo scroll in qualsiasi direzione
Istruzioni per la gestione dei requester	
REQCOLOR	Altera il colore del testo di un requester
SREQCOLOR	Altera il colore del testo di un requester stringa personalizzato
Istruzioni per la gestione dei file	
DISKSPACE	Riporta lo spazio libero e quello utilizzabile di un disco
DOS	Esegue un comando AmigaDOS dall'interprete Basic
EXISTS	Controlla se esiste un certo file nella directory corrente o in quella specificata
FILESIZE	Riporta le dimensioni di un file in byte
GETFILES	Apre un requester contenente i nomi dei file su disco e attende una scelta dell'utente
Altre istruzioni	
BITPOS	Restituisce le coordinate del puntatore nella bit map
BUSY	Trasforma il puntatore del mouse nella parola "WAIT"
CREDITS	Schermata di citazioni e ringraziamenti per la realizzazione di Extend
FONT	Specifica la fonte-carattere e le sue dimensioni, per l'uso nella finestra del Basic
LOADIFF	Carica in memoria e mostra uno schermo salvato in formato IFF
PROTECT	Protegge una stringa Basic da alterazioni
PTIME	Lascia passare un po' di tempo affinché il Basic non si blocchi
SAVEIFF	Salva su disco una schermata in formato IFF
SETSTYLE	Imposta gli attributi del testo facendo uso di un numero per lo stile
VERSION	Riporta la versione della libreria "extend.library" in uso
VIEVHAM	Porta sullo schermo un'immagine in modo HAM (Hold and Modify)

gramma in linguaggio C.

Il secondo gruppo racchiude le funzioni per i gadget, che consentono la definizione e l'uso dei gadget di Intuition nei propri programmi.

Il terzo gruppo comprende le funzioni relative al trattamento dei menu: come si può notare, attraverso queste funzioni è possibile definire con una buona precisione la struttura di menu e sotto-menu da utilizzare nei propri programmi, nonché rispondere in maniera adeguata alle interazioni che l'utente compie con i menu.

Troviamo poi le due funzioni per il trattamento dinamico della memoria: MEMORY e FREEMEM servono a richiedere e rilasciare memoria di sistema, all'incirca nello stesso modo in cui si opererebbe in un programma in C tramite la funzione AllocMem.

Esistono inoltre i comandi per lo scroll delle bitmap negli schermi del Basic, che consentono anche a programmi scritti in questo linguaggio di far uso di tecniche esclusive dell'Amiga, aprendo quindi la porta a giochi di stile arcade.

Il gruppo successivo di funzioni è delegato alla gestione dei requester, in particolare per quanto riguarda i testi di un requester stringa.

L'ottava categoria contiene comandi che permettono di gestire i file AmigaDOS e di mandare in esecuzione i comandi del CLI. In particolare, un'istruzione permette di visualizzare un requester tramite il quale l'utente, impiegando il mouse, può accedere alle directory dei dischi e selezionare file, il tutto senza che il programma Basic debba chiamare una sola funzione di sistema.

L'ultimo insieme di comandi descritto nella tavola è costituito da 11 funzioni miste, due delle quali servono a manipolare file in formato IFF e un'altra per visualizzare immagini in modo HAM da Basic. I tre esempi che seguono in questo articolo possono servire per comprendere meglio il funzionamento e l'utilità di ognuna di queste funzioni; questi esempi sono tratti dalla directory s del disco sistema di Extend.

Il primo programma

Per il primo esempio facciamo riferimento al Listato 1. Questo programma mostra come si possono costantemente tenere sotto controllo diversi gadget per stabilire quale di essi è stato selezionato dall'utente. In questo programma si fa uso delle istruzioni SetWindow, GadTColor, Gadget, OffGad, OnGad, Refresh, RemoveGad e CloseLib, e soprattutto viene illustrata l'utile l'istruzione Check.

Se su uno schermo Amiga BASIC vi sono diversi gadget di Intuition, tutti creati attraverso la funzione Gadget, Check restituisce il numero del gadget selezionato dall'utente, sempre che tale numero sia compreso all'interno della gamma di valori indicata nei suoi argomenti. Nell'esempio si fa riferimento a un solo gadget, ma non è difficile renderlo più generico.

La prima istruzione del programma definisce la variabile q e il valore restituito dalla funzione Check come numeri interi di tipo long, ossia formati da quattro byte. Poi viene aperta la libreria extend.library nella solita maniera, e si dichiara la funzione di libreria Check come una funzione e non come un'istruzione (la differenza è che una funzione restituisce un valore); la successiva istruzione SetWindow utilizza la funzione WINDOW per passare alle altre funzioni della extend.library l'indirizzo base della struttura che definisce la finestra di output del Basic.

Con le altre istruzioni si definiscono le caratteristiche del gadget, come il tipo e la forma (per chi ha conoscenze di C, diciamo che queste caratteristiche vengono organizzate seguendo la disposizione dei parametri nella struttura Gadget di Intuition); la funzione GadTColor serve a identificare le modalità di disegno del gadget, cioè i colori di primo piano e di sfondo e il modo grafico. In seguito si attiva il gadget tramite la funzione Gadget e si usano le istruzioni standard PRINT e LOCATE per collocare un testo al suo interno. La funzione Gadget utilizza tre parametri: il numero del gadget, il suo nome e i suoi attributi.

In seguito si fa uso di tre istruzioni per stabilire se l'utente ha selezionato il gadget. Come si può vedere, la funzione Check ha bisogno di due parametri: il numero del primo gadget da controllare e il numero dell'ultimo. Il valore restituito dalla funzione finisce nella variabile c, che assume il valore 1 quando il gadget è stato selezionato. A questo punto si usa il comando OffGad per disabilitare il gadget, e si attende che l'utente prema il tasto RETURN (attraverso l'istruzione INPUT x).

Quindi si usa la funzione OnGad per riattivare il gadget precedentemente disabilitato: viene cancellata la finestra di output con l'istruzione CLS e il gadget viene ridisegnato mediante le consuete istruzioni LINE. Successivamente, il programma esegue l'istruzione Refresh dell'estensione all'Amiga BASIC per effettuare il refresh della finestra. A questo punto il programma, dopo aver visualizzato all'interno del gadget un messaggio per l'utente, si ferma nuovamente, in attesa che l'utente selezioni il gadget. La variabile c viene nuovamente azzerata e un nuovo ciclo chiama la funzione Check e attende il responso dell'utente. Alla fine, quando l'utente preme il pulsante sinistro del mouse sul gadget provoca i seguenti effetti: il gadget viene cancellato, va in esecuzione l'istruzione CloseLib per liberare tutta la memoria utilizzata dalla struttura Gadget, viene chiusa la libreria extend.library e viene cancellata la finestra di output del Basic.

Il secondo esempio

Nel Listato 2 viene illustrato l'uso delle funzioni SetWindow, CheckSub, PTime, SUBCOLOR, SUBMENU e COMMANDKEY. In particolare si esami-

na come va impiegata la funzione CheckSub, che non ha bisogno di alcun argomento ma restituisce il numero relativo al sotto-menu selezionato dall'utente; nell'esempio, questo numero viene memorizzato nella variabile submenu.

Dapprima il programma esegue una serie di operazioni analoghe a quelle dell'esempio precedente, usando le istruzioni DEFLNG, LIBRARY e DIM e la funzione SetWindow di Extend, che servono rispettivamente ad aprire le librerie necessarie, dichiarare le variabili (strutture) usate e indicare all'ambiente Extend qual è la finestra Basic di output in uso. In seguito, una serie d'istruzioni standard MENU specificano due menu con quattro voci, collegati alla finestra del Basic.

L'istruzione che segue, PTime, fa fermare il programma per 50 tick, ossia per un secondo, allo scopo di dare all'Amiga BASIC e a Intuition il tempo necessario per effettuare il refresh dello schermo: in tal modo quando il programma riprende la propria esecuzione la finestra avrà raggiunto il suo aspetto definitivo. L'istruzione SubColor imposta il colore e il modo grafico per le voci dei menu da disegnare; questa istruzione è simile all'istruzione COLOR dell'Amiga BASIC.

Le successive sei istruzioni SUBMENU definiscono le voci dei sotto-menu, indicando rispettivamente il numero del menu, il numero della voce, il numero della voce nel sotto-menu e il nome del sotto-menu. Viene usata la funzione standard del Basic SADD (String ADDress) per conoscere l'indirizzo in memoria delle stringhe di caratteri.

L'istruzione COMMANDKEY serve invece ad assegnare un tasto a una voce di un menu: in tal modo l'opzione può essere richiamata attraverso quel tasto (o combinazione di tasti) oltre che attraverso l'uso del mouse. Questo comando ha bisogno di quattro parametri: il numero del menu, il numero della voce, il numero del sotto-menu e il carattere del tasto corrispondente. Nel nostro esempio, alla prima voce viene collegato il tasto "Q", alla seconda il tasto "S" e alla terza il tasto "4".

Le ultime due istruzioni consistono di istruzioni MENU dell'Amiga BASIC che vengono utilizzate per ottenere il numero dell'ultimo menu o dell'ultima voce scelta dall'utente; queste informazioni vengono memorizzate rispettivamente nelle variabili "men" e "it" del programma.

Quando poi avviene una selezione, la funzione CheckSub dice al programma quale voce del sotto-menu è stata selezionata e il programma ne indica il numero all'utente attraverso le istruzioni Basic standard LOCATE e PRINT. Se infine viene attivata la terza opzione del primo menu, il programma cancella l'albero dei menu e termina.

Il terzo esempio

Nel terzo esempio vediamo l'uso di Check, SetWindow, Gadget, GadTColor, GadBColor, Putstring, Refresh, RemoveGad e CloseLib. Oltre a mostrare l'impiego di due gadget, in questo programma si mettono in luce le modalità d'uso della funzione Putstring. Come di consueto, la prima parte del programma è analoga alle precedenti e svolge le consuete dichiarazioni oltre ad aprire la libreria di Extend e a passarle il puntatore alla finestra attiva del Basic; in seguito vengono anche definite le linee costituenti i gadget e le caratteristiche del primo gadget vengono indicate mediante le funzioni Gadget, GadTColor e GadBColor.

Il primo gadget è senza bordo; il secondo è invece booleano e il programma riserva un buffer per il testo a esso relativo, testo che verrà inserito al suo interno mediante la funzione Putstring. Il programma non fa altro che dire all'utente di selezionare il gadget booleano per vedere i risultati prodotti dall'istruzione Putstring nella finestra di output; a questo punto, il programma libera la memoria occupata e termina.

Conclusioni

Se si vogliono estendere le potenzialità dell'Amiga BASIC e sfruttare con facilità le caratteristiche di Intuition, Extend è indispensabile. Il tool viene fornito insieme con programmi dimostrativi che illustrano le modalità d'impiego di ognuna delle 47 nuove funzioni; questi programmi possono essere utilizzati sia dal CLI sia dal *Workbench*. Extend è dedicato soprattutto a quei programmatori che vogliono ottenere risultati di alto livello senza doversi preoccupare d'imparare il linguaggio C.

Osserviamo infine che, grazie alla struttura di Extend, i programmi che impiegano i nuovi comandi possono essere compilati con il compilatore AC/BASIC della Absoft.

Al modico prezzo di 39 dollari il pacchetto Extend può essere ordinato presso:

Sunsmile Software
533 Fargo Avenue
Buffalo, NY 14213, USA
(Tel. 001/716/8855670)

Nel prossimo articolo

Nel prossimo articolo di questa serie parleremo dell'editor *TxEd Plus* (da poco disponibile nella versione 2.00 dalla MicroSmiths) e delle utility che lo accompagnano: BlitzDisk, FastFonts, FunKeys. Si tratta di un word processor potente e facile da usare, che insieme a *CygnusEd Professional* costituisce uno dei più funzionali editor in commercio.

TxEd Plus rappresenta una scelta meno sofisticata, pur avendo diverse caratteristiche pregevoli sia per la stesura di programmi sia per la scrittura di testi. È senz'altro molto più elementare di *CygnusEd*, ma consente ugualmente di personalizzare i menu e la tastiera e presenta anche la compatibilità AREXX. Oltre 50 comandi possono essere inviati alla sua porta AREXX da altri task, e l'editor può dunque essere controllato integralmente dall'esterno. La velocità e le ridotte dimensioni sono

altre caratteristiche che lo rendono prezioso per alcuni programmatori.

Per quanto riguarda le utility fornite a corredo, diciamo subito che BlitzDisk, come dice il nome, serve per rendere più veloci gli accessi al disco; FastFonts serve per migliorare la gestione dei caratteri; FunKeys fornisce un valido metodo per creare hot key personalizzate. Il tutto per 79 dollari.

Listato1: *la funzione Check*

```
DEFLNG q,Check
LIBRARY "extend.library"
DECLARE FUNCTION Check() LIBRARY
SetWindow WINDOW(7)
DIM array%(9)
LINE (159,40)-(159+116,40+64),2,bf
LINE (159,40)-(159+112,40+60),1,bf
array%(1)=0
array%(2)=159
array%(3)=40
array%(4)=112
array%(5)=60
array%(6)=1
array%(7)=0
array%(8)=1
array%(9)=20
GadTColor 3,1,0
q=VARPTR(array%(1))
Gadget 1,$ADD("CHECK example" + CHR$(0)),q
LOCATE 1,1:PRINT "Click to disable Gadget"
WHILE c<>1
 c=Check(1,1)
WEND
OffGad 1
PRINT "Press <RETURN>"
INPUT x
OnGad 1:CLS:LINE (159,40)-(159+119,40+64),2,bf
Refresh WINDOW(7)
PRINT "Click to remove Gadget"
c=0
WHILE c<>1
 c=Check(1,1)
WEND
RemoveGad 1
CloseLib
LIBRARY CLOSE
CLS
END
```

Listato 2: *la funzione CheckSub*

```
DEFLNG a-z
LIBRARY "extend.library"
SetWindow WINDOW(7)
DECLARE FUNCTION CheckSub() LIBRARY
MENU 1,0,1,"First Menu"
MENU 1,1,1,"   Menu 1, item 1"
MENU 1,2,1,"   Menu 1, item 2"
MENU 1,3,1,"   Menu 1, item 3"
MENU 1,4,1,"   Menu 1, item 4"
MENU 2,0,1,"Second Menu"
MENU 2,1,1,"   Menu 2, item 1"
MENU 2,2,1,"   Menu 2, item 2"
MENU 2,3,1,"   Menu 2, item 3"
MENU 2,4,1,"   Menu 2, item 4"
PTime 50
SUBCOLOR 2,0,0
SUBMENU 1,3,1,$ADD("MENU 1, item 3,sub one"+
CHR$(0))
SUBMENU 1,3,2,$ADD("MENU 1, item three, (QUIT)"+
CHR$(0))
SUBMENU 2,2,1,$ADD("MENU 2, item 2,sub one"+
CHR$(0))
SUBMENU 2,2,2,$ADD("MENU 2, item 2,sub 2"+CHR$(0))
SUBMENU 2,2,3,$ADD("MENU 2, item 2,sub 3"+CHR$(0))
SUBMENU 2,2,4,$ADD("MENU 2, item 2,sub 4"+CHR$(0))
PRINT "Select Quit to quit demo"
COMMANDKEY 1,3,2,$ADD("Q")
COMMANDKEY 1,3,1,$ADD("S")
COMMANDKEY 2,2,4,$ADD("4")
10 men=MENU(0):it=MENU(1)
submen=CheckSub:LOCATE 10,10:PRINT"Sub #=";submen
IF men=1 AND it=3 AND submen=2 THEN MENU RESET
END
GOTO 10
```

Listato 3: *la funzione Putstring*

```
DEFLNG q,Check
LIBRARY "extend.library"
DECLARE FUNCTION Check() LIBRARY
SetWindow WINDOW(7)
DIM array%(9)
LINE (259,20)-(259+244,20+11),2,bf
array%(1)=2
array%(2)=259
array%(3)=20
array%(4)=240
array%(5)=8
array%(6)=4
array%(7)=40
array%(8)=83
array%(9)=1
q=VARPTR(array%(1))
Gadget 1,$ADD("Type here:"+CHR$(0)),q
array%(1)=0:array%(2)=100:array%(3)=20
array%(4)=60:array%(5)=10:array%(6)=1
array%(7)=0:array%(8)=100:array%(9)=1
GadTColor 2,1,1
GadBColor 0,0,0 "a no border Gadget"
q= VARPTR(array%(1))
Gadget 2,$ADD("Boolean"+CHR$(0)),q
PRINT "Click boolean to see PUTSTRING"
GOSUB hang
buf$="This was put here!"+CHR$(0)
Putstring 1,$ADD(buf$)
Refresh WINDOW(7)
PRINT "Click boolean to end"
GOSUB hang
RemoveGad 1
RemoveGad 2
CloseLib
LIBRARY CLOSE
CLS
END

hang:
c=0
WHILE c=0
 c=Check(2,2)
WEND
RETURN
```

■

ABBONATI!

A CASA TUA UN REGALO OGNI MESE

ABBONARSI CONVIENE!

- Sicurezza di non perdere neanche un numero
- Prezzo bloccato per un anno
- Sconto del 15% sul prezzo di copertina

- Possibilità d'includere nell'abbonamento gli arretrati
- Comodità di ricevere la rivista direttamente a casa

Per chiunque sia interessato ai computer Commodore, *Commodore Gazette* è indispensabile. Nessuna rivista in Italia offre ai suoi lettori tanta qualità con recensioni hardware e software, listati, presentazioni esclusive, informazioni di ogni genere riguardanti C-64, C-128 e Amiga.

Un abbonamento a *Commodore Gazette* è il regalo più bello che possiate fare a voi stessi e agli altri... un regalo nuovo ogni mese. Ma non è finita! Risparmierete il 15% sul prezzo di copertina e potrete includere nel prezzo dell'abbonamento anche i numeri arretrati che mancano alla vostra raccolta. Resta inteso che per ogni arretrato scelto verrà spedito un numero di prossima uscita in meno. Per esempio, chi si abbona a 12 numeri a partire dal n. 1/89 e richiede 5 arretrati, riceverà 7 numeri del 1989 e 5 arretrati.

Ritagliare e spedire a: IHT Gruppo Editoriale – Via Monte Napoleone, 9 – 20121 Milano

Sì, desidero sottoscrivere un abbonamento a ☐ 12 numeri (L. 81.600) ☐ 24 numeri (L. 163.200) di COMMODORE GAZETTE usufruendo così dello sconto del 15% sul prezzo di copertina

Nome e cognome .....

Indirizzo ..... Città ..... CAP .....

Inizio abbonamento dal n. ..... Desidero ricevere i seguenti numeri arretrati (specificare numero e anno) .....

☐ Allego assegno bancario o circolare ☐ Allego fotocopia della ricevuta del vaglia postale

Firma .....

L'USO DEGLI INTERRUPT PER LE TEMPORIZZAZIONI DEI TASK

Uno studio degli interrupt relativi ai CIA dell'Amiga ci consente di utilizzarli per temporizzare le procedure all'interno dei task. Un listato in C e in Assembly ne mostra un impiego reale

di Eugene P. Mortimore

Spesso sorge la necessità di utilizzare temporizzazioni specifiche nei propri programmi: può esserci l'esigenza di far eseguire alcune routine in un certo ordine e/o di ritardarne nel tempo l'esecuzione, cioè creare un'organizzazione nella quale un task esegue una routine, entra in attesa per un certo tempo, riottiene il controllo ed esegue una seconda routine, rientra in attesa, esegue una certa routine... e così via.

Si può desiderare che questi intervalli di attesa siano rigorosamente prestabiliti in frequenza e durata, oppure che siano casuali; nel secondo caso la frequenza di ripetizione e la durata delle attese può per esempio dipendere da numeri casuali generati dal computer o da eventi esterni come l'azione dell'utente sull'interfaccia grafica Intuition. In questo articolo vedremo com'è possibile impiegare gli interrupt generabili dall'interfaccia hardware 8520 (CIA, Complex Interface Adapter) per pianificare vere e proprie "strategie di temporizzazione" che ci consentano di creare flussi di controllo come quelli appena descritti.

Come usare gli interrupt dei timer

Gli interrupt generati dall'hardware costituiscono un aspetto fondamentale del meccanismo hardware-software che interfaccia i programmi con il mondo esterno (le porte parallela e seriale, per esempio) e con i disk drive. Il concetto di interrupt e il modo di servirsene sono argomenti tanto importanti che abbiamo dedicato il nostro primo articolo (sul numero 5/88) all'analisi del loro comportamento e della loro gestione.

Ora svolgeremo una trattazione specifica degli interrupt hardware dell'Amiga per capire come si possono usare per generare "ritardi" all'interno dei propri task. Prima, però, apriamo una breve parentesi per considerare più in generale il concetto di temporizzazione nell'Amiga, un soggetto che diventa molto complesso quando si scopre che ci sono molti livelli software da cui può essere controllata la temporizzazione dei programmi. In primo luogo esiste il controllo diretto dei registri hardware relativi agli interrupt, se un programmatore è abbastanza pratico dell'hardware dell'Amiga da saper manipolare direttamente i singoli bit dei registri hardware. Queste persone si trovano nella posizione migliore in quanto riescono a controllare la loro macchina con la massima efficienza, ma devono agire con estrema cautela dal momento che senza le opportune precauzioni un approccio così diretto può generare disastri in una macchina multitasking, dove anche le risorse hardware sono condivisibili fra più task.

Poi esistono alcuni programmatori che si trovano perfettamente a loro agio con il linguaggio C e hanno imparato a trattare con i singoli bit nei modi consentiti da questo linguaggio. Ma anche se sono in molti a conoscere il C, in genere si preferisce non usarlo per un controllo della macchina di così basso livello in quanto spesso si scopre che il grado di precisione necessario supera le proprie conoscenze oppure che un accurato debug dei programmi che accedono direttamente ai registri hardware è difficilissimo. Tuttavia, a mano a mano che ci si allontana dal livello più basso di manipolazione, quello offerto dal C o dall'Assembly, si perde la possibilità di definire con precisione le temporizzazioni dei propri programmi.

Altri programmatori si servono del C ma preferiscono usare solo l'interfaccia di Intuition per controllare le temporizzazioni dei propri programmi. Si tratta di un metodo abbastanza flessibile, ma implica l'uso del dispositivo Timer:

questo limita in qualche modo il nostro approccio a causa dell'intrinseca imprecisione delle funzioni di quel dispositivo, che aumenta con l'aumentare del numero di task attivi nel sistema.

Le varie possibilità

Le necessità di temporizzazione dei programmi si dividono sempre in due categorie: attesa di eventi provenienti dall'esterno e temporizzazioni che dipendono da valori predeterminati o da calcoli interni al task.

In generale, le temporizzazioni che si basano su eventi esterni richiedono una particolare gestione dei segnali dell'Exec relativi agli interrupt hardware, mentre quelle generate dai task devono far uso di segnali dell'Exec relativi a interrupt hardware generati in seguito al raggiungimento di determinati valori da parte di contatori hardware del CIA. Tutto questo risulterà più chiaro nell'esempio che descriveremo in seguito.

Il sistema Exec fornisce almeno sei funzioni – Signal, AllocSignal, FreeSignal, WaitPort, Wait e WaitIO – direttamente correlate ai segnali software dell'Amiga. Tramite queste funzioni è possibile scrivere un programma che alloca un segnale dell'Exec, quindi entra in attesa e infine riprende il controllo quando il segnale viene effettivamente generato. Il segnale può provenire dal sistema e da altri task. Nella nostra trattazione supponiamo che i segnali vengano generati da un'opportuna routine che allochiamo perché riceva il controllo a ogni interrupt. Ovviamente, questa routine di interrupt può anche "risvegliare" il programma ogni dieci o ogni cento interrupt, a seconda delle proprie necessità.

Quando riottiene il controllo, il programma può iniziare a svolgere i compiti per cui è stato progettato. Questo è il meccanismo usato più spesso per collegare i segnali dell'Exec agli interrupt hardware. Nel programma d'esempio che accompagna l'articolo viene anzitutto installata una routine di interrupt che ottiene il controllo ogni volta che il Timer-A del CIA genera un interrupt (evento che si svolge ogni volta che il contatore del timer raggiunge lo zero), e viene allocato un segnale del quale la funzione principale main entra periodicamente in attesa. Ogni volta che si verifica un interrupt, la routine di gestione riceve il controllo e invia un segnale alla funzione main in attesa, la quale riprende l'esecuzione e chiama una particolare funzione di output a video. È importante capire che questo rapporto di causa-effetto non è automatico ma è una precisa scelta del nostro programma. Analoghi meccanismi vengono utilizzati internamente dal sistema Exec per dare origine ai suoi diversi task. Analizzando il funzionamento di questo programma si comprende la dipendenza tra interrupt hardware e segnali software.

Salendo all'approccio di livello superiore, il dispositivo Timer mette a disposizione tre funzioni, AddTime, SubTime e CmpTime, che possono essere usate per gestire intervalli di tempo in tre modi: sommandoli, sottraendoli e confrontandoli. Poiché sono fornite direttamente da un dispositivo dell'Amiga, queste tre funzioni possono essere usate nell'ambiente multitasking in modo che un task possa controllare le operazioni di un altro e viceversa. Gli interrupt del CIA, invece, possono essere utilizzati solo da un task alla volta, come si spiega più avanti.

Al livello ancora superiore, Intuition ci mette a disposizione una sola funzione per la gestione delle temporizzazioni. CurrentTime restituisce l'ora corrente in secondi e microsecondi e lascia al programmatore il compito di utilizzare questi dati per far partire l'esecuzione di certe routine nei suoi programmi. Quindi, usata da sola, quest'opportunità non consente un'interazione diretta con il mondo esterno ed è utile solo per generare temporizzazioni basate su calcoli interni, e non su reali attese.

Salendo ulteriormente di livello, per creare temporizzazioni possiamo impiegare alcune istruzioni del Basic, dell'AmigaDOS, e perfino del CLI.

In questa ascesa si scopre che a ogni ulteriore livello diminuisce la capacità di ricevere, riconoscere e agire in base a determinati eventi. Questo ci porta alla generica conclusione che un preciso controllo degli eventi esterni e della loro controparte interna (nonché una precisa temporizzazione delle routine dipendenti da questi eventi) richiede l'uso delle funzioni dell'Exec relative ai segnali o delle funzioni del dispositivo Timer. Si noti, a proposito di quest'ultimo dispositivo, che nonostante sia facile usarlo in un sistema multitasking, le tre sole operazioni messe a disposizione sono piuttosto limitate, soprattutto in un sistema sovraffollato.

Siamo dunque costretti a prendere in considerazione l'intervento diretto sui registri hardware dei chip per scoprire che cosa ci offrono, arrivando finalmente all'argomento principale di questo articolo: l'accesso diretto e la manipolazione dei registri del circuito integrato 8520 allo scopo di ottenere temporizzazioni precise nei propri programmi. Poiché si tratta di un argomento complesso, specialmente in un sistema multitasking come l'Amiga, divideremo la trattazione in due fasi.

In questa prima parte discuteremo in modo generico le caratteristiche dei due CIA dell'Amiga, e le caratteristiche di massima di un programma che agisce sui singoli registri in essi presenti per ottenere precise temporizzazioni. Nel prossimo articolo analizzeremo più a fondo le operazioni e le caratteristiche di questo programma. Inoltre prenderemo in considerazione il problema più generale di progettare un insieme di routine che interfacciano il software sistema con l'hardware dell'Amiga. Il risultato sarà un importante punto di partenza per sviluppare programmi personalizzati che sfruttano fino in fondo le potenzialità chip hardware.

Descrizione del chip 8520

L'Amiga utilizza due CIA 8520, ai quali ci riferiremo chiamandoli 8520-A e 8520-B. Questi circuiti traggono il loro nome (Complex Interface Adapter) dal fatto che forniscono una complicata interfaccia fra la CPU e il mondo esterno. Il loro scopo è convertire i segnali elettrici che provengono da svariate fonti, come la porta seriale e quella parallela, in segnali logici (principalmente interrupt) che il 68000 può comprendere e utilizzare. Inoltre possono essere utilizzati per generare impulsi elettrici o interrupt anche internamente, e il già citato contatore interno che arriva a zero è solo uno dei possibili esempi. Questi due chip sono virtualmente identici tranne che per alcuni particolari che non riguardano la nostra discussione. Non è nostra intenzione descrivere una per una tutte le caratteristiche dei chip 8520, ma è necessario soffermarci almeno su quelle che riguardano il breve programma di temporizzazione che stiamo per discutere. Un elenco completo, anche se piuttosto oscuro, di tutte queste funzionalità è riportato nell'appendice F dell'*Amiga Hardware Reference Manual*.

La Figura 1 mostra un diagramma a blocchi del dispositivo 8520. I "segnali di controllo del chip", nella parte bassa della figura, sono i segnali elettrici utilizzati per consentire al software l'accesso al circuito integrato e il suo controllo. In particolare, le linee indicate rappresentano rispettivamente un indicatore di Lettura/Scrittura (linea R/$\overline{W}$), il clock di sistema, corrispondente a una delle due fasi del clock (linea Φ2), una linea per l'abilitazione del chip (linea $\overline{CS}$), per il ripristino (linea $\overline{RES}$) e per l'indirizzamento dei registri (linee RS0...RS3).

Le ultime quattro linee sono importanti in quanto ci consentono d'indirizzare i registri interni del chip e di compiervi operazioni di lettura o scrittura: si tratta di 16 registri a otto bit. Le operazioni di lettura o di scrittura vengono selezionate attraverso l'invio di uno stato logico 1 oppure 0 sulla linea R/$\overline{W}$. Si noti che alcuni registri sono a sola lettura e altri sono a sola scrittura. Le funzioni di libreria della risorsa CIA che descriveremo in seguito obbediscono alle regole di scrittura e lettura dei singoli registri. Ogni bit di ogni registro ha un nome e una funzione specifica che può essere utilizzata direttamente o attraverso una routine di sistema. Si faccia ancora riferimento all'*Amiga Hardware Reference Manual* per quanto riguarda l'elenco e la descrizione dei registri.

Il resto di questo diagramma riporta i registri buffer e la maggior parte degli effettivi registri del chip 8520. Si noti in particolare il registro chiamato "ICR/MASK" in basso a sinistra nella Figura 1. In questo registro giungono cinque diverse linee elettriche provenienti da cinque diversi registri denominati rispettivamente FLAGBUFFER, SERIALPORT, TODALARM, Timer-A e Timer-B. Come gli altri, anche questo è un registro a otto bit e presto ne analizzeremo i flag più importanti per la nostra discussione. Le routine software relative, come SetICR e AbleICR della risorsa di sistema cia.resource, possono leggere e scrivere in ciascun bit di questo registro.

Come abbiamo visto, ciascuno dei due chip 8520 è in grado di generare cinque diversi tipi di interrupt. Si verificano due diversi interrupt della porta seriale, sia quando il buffer interno è pieno (RBF, buffer di ricezione pieno), sia quando il buffer è vuoto (TBE, buffer di trasmissione vuoto).

L'interrupt TODALARM si verifica quando l'orologio "time-of-day" (TOD) dell'8520 raggiunge l'ora che un task ha impostato per l'allarme. Questo orologio è a 24 bit (impegna tre registri del chip). Questi registri non verranno più presi in considerazione nella nostra discussione, e nemmeno il successivo, FLAGBUFFER.

La linea IRQ – Interrupt ReQuest – proveniente dal registro IRQ BUFFER giunge al chip Paula come si può vedere nelle Figure 1 e 2. In particolare, la linea IRQ proveniente dal chip 8520-A entra nell'ingresso a bassa priorità EXT-2 (interrupt di livello 2) del chip Paula, mentre la linea IRQ del secondo CIA si collega all'ingresso ad alta priorità EXT-6.

Il circuito integrato Paula decodifica tutti gli interrupt generati da ciascun CIA cosicché ognuno dei possibili dieci interrupt venga inserito nell'esatta posizione (livello e sotto-livello) dello schema degli interrupt dell'Amiga. Si rammenta che l'Amiga riconosce 15 livelli di priorità degli interrupt e che questi livelli sono raggruppati in sette sotto-livelli. In questo modo, anche se il 68000 può riconoscere e gestire solo sette livelli di interrupt, il chip Paula può arrivare fino a 15 traducendoli, attraverso uno schema di decodifica, nei sette livelli contemplati dalla CPU (si noti che il livello 7 non viene mai usato all'interno dell'Amiga; è attivabile soltanto attraverso un pin del connettore d'espansione e serve soprattutto nel debug).

Tutti i 15 tipi di interrupt ammessi dal chip Paula e i relativi sottolivelli sono discussi nell'*Amiga ROM Kernel Reference Manual* e in alcuni precedenti articoli di questa serie, in particolare quelli dove si è trattata la struttura ExecBase e i suoi parametri.

In aggiunta a tutto questo, senza dipendere ulteriormente dal mondo esterno, entrambi i chip sono capaci di generare propri segnali di interrupt e inviarli alla CPU attraverso una delle due linee connesse al chip Paula. È in questo modo che i due CIA possono agire come fonti esterne di interrupt, ed è in questo modo che vengono usati come generatori di temporizzazioni. Per quest'ultimo impiego, di solito si comincia chiamando una routine che inizializza via software il conto alla rovescia dei loro contatori (che fanno capo ai registri TALO e TAHI), e si fa in modo che il CIA impieghi il clock di sistema per decrementare il contatore. Poi, quando il contatore raggiunge lo zero, il chip invia un impulso di interrupt alla CPU

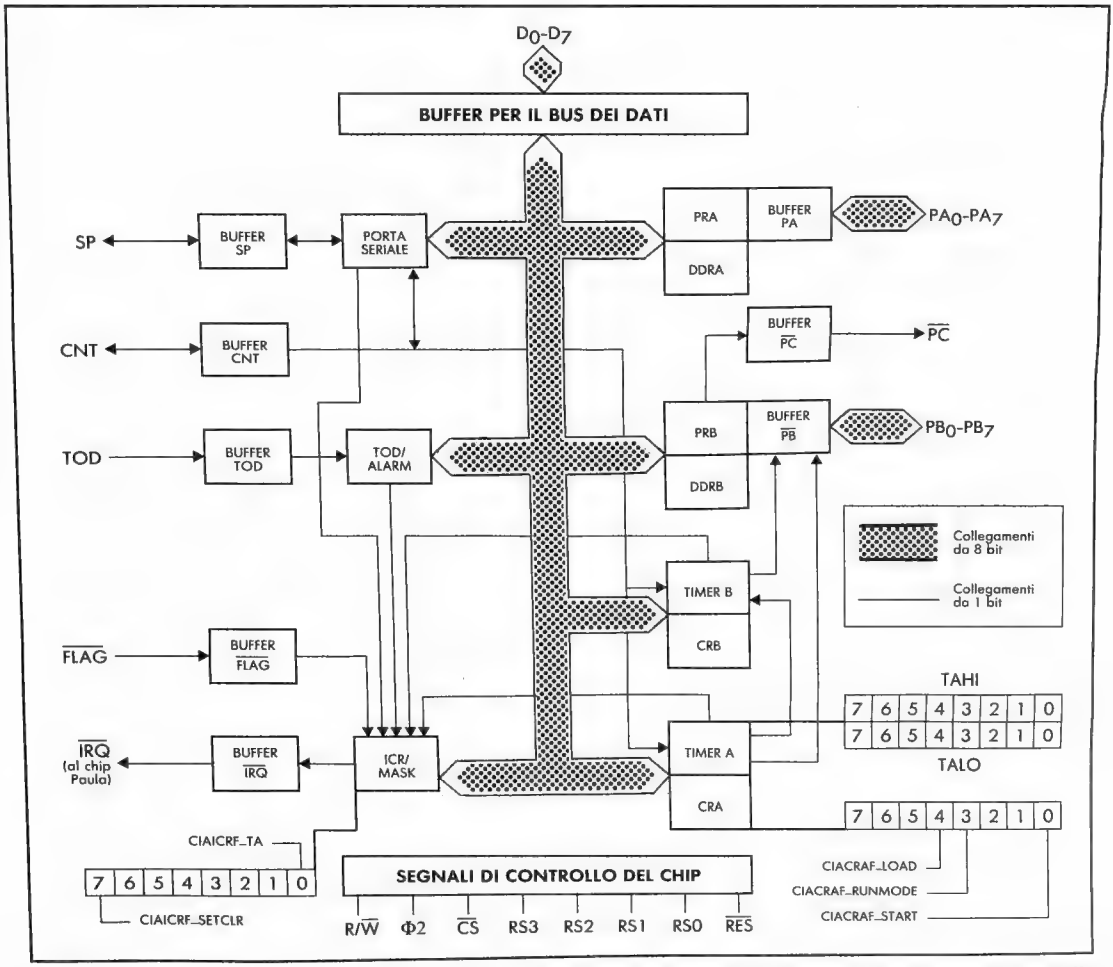


Figura 1: schema a blocchi del chip 8520 (CIA) con i nomi dei registri dei singoli bit. Si notino in basso i segnali di controllo

e questo causa l'esecuzione della relativa routine di gestione.

Ora prendiamo in considerazione gli impulsi di interrupt generati dai due timer (Timer-A e Timer-B) e incanalati attraverso le relative linee nel registro ICR/MASK, come mostra la Figura 1. Si possono usare gli interrupt di questi due timer sia separatamente sia in combinazione, per produrre i tipi di temporizzazioni richieste dai propri programmi. Tuttavia si deve tener conto del fatto che gli interrupt dei due timer del chip 8520-B sono perlopiù riservati al sistema.

Come il sistema utilizza gli interrupt dei timer

Il Timer-A dell'8520-B è usato per la gestione dei dati in arrivo dalla tastiera quando le routine dell'Exec sono attive; il Timer-B, invece, è usato per generare gli interrupt relativi alla gestione multitasking del sistema, interrupt che di solito avvengono 16 volte al secondo. Entrambi questi interrupt sono dunque continuamente utilizzati dal sistema e l'unico caso in cui questo non accade è quando qualcuno prende il controllo della macchina durante il boot prima delle routine dell'Exec: poiché questa è una situazione molto rara, in pratica i due timer non vengono mai usati dal programmatore.

Similmente, gli interrupt del Timer-B del chip 8520-A vengono usati come "inseguitori" del pennello elettronico, ossia per sincronizzare il chip Blitter con la posizione del fascio di elettroni che genera l'immagine sul video. Dal momento che l'Amiga è un sistema multitasking e molti task possono fare uso contemporaneamente del Blitter per spostare dati più velocemente, non vi è un metodo semplice per sapere se altri task stanno usando il Blitter. Quindi, per sicurezza è sempre

meglio utilizzare per le nostre esigenze di temporizzazione il solo timer rimasto libero.

Il Timer-A dell'8520-A viene usato per la porta seriale, ma solo quando ha luogo un trasferimento di dati; ciò ne consente l'impiego quando nessun task sta facendo uso della porta seriale. Naturalmente, bisogna assicurarsi che nessuna comunicazione sulla porta seriale possa iniziare durante l'uso del Timer-A da parte del nostro task.

Il resto della discussione sarà limitato all'uso dei registri del Timer-A dell'8520-A per generare le temporizzazioni nel nostro programma d'esempio. Faremo quindi conoscenza con i seguenti registri del timer: CRA (registro di controllo del Timer-A), TALO e TAHI (rispettivamente, i registri per l'impostazione del byte basso e del byte alto del valore d'inizio del conto alla rovescia). La Figura 1 illustra in quale posizione sono collocati questi registri.

Per prima cosa dovremo fare in modo che il nostro programma inizializzi il registro di controllo degli interrupt del chip 8520: il registro ICR/MASK. Questo è necessario per azzerare il bit del Timer-A che potrebbe essere impostato quando il nostro programma prende il controllo. Un'altra operazione di scrittura è necessaria per abilitare l'interrupt e il conto alla rovescia del Timer-A.

La Figura 2 mostra esplicitamente come i registri del Timer-A influenzano le operazioni del sistema. Questa figura richiama l'attenzione sui quattro registri chiamati in causa nel programma che ora descriveremo. Per indicare che il chip 8520-A verrà usato per effettuare un conto alla rovescia nel suo Timer-A, il programma deve scrivere gli opportuni valori nel registro ICR/MASK, nel registro TAHI, in TALO e nel registro CRA.

Una volta che questi valori sono stati scritti e che il Timer-A è stato attivato da un'ulteriore scrittura nel registro CRA, questo contatore inizia immediatamente a decrementarsi e quando il conteggio raggiunge lo zero il chip genera automaticamente un impulso di interrupt. Questo impulso passa dal registro del Timer-A al registro ICR/MASK, al chip Paula e infine al 68000. Successivamente, il controllo viene ceduto all'Exec.

A questo punto la routine di interrupt hardware che il nostro programma aggiunge al sistema riceve il controllo, incrementa la variabile globale Timer e chiama la funzione Signal dell'Exec per inviare un segnale alla parte del programma che è entrata in attesa tramite la funzione Wait.

Questo processo continua per un certo numero di volte (si rammenti che il contatore del timer è stato programmato per ripartire automaticamente quando raggiunge lo zero). Per quanto riguarda le operazioni di scrittura e lettura nei registri del CIA, faremo uso delle quattro funzioni create per la gestione della risorsa hardware CIA che nel prossimo paragrafo descriveremo in ogni particolare.

Le funzioni relative al CIA

Esaminando il listato che accompagna l'articolo si notano quattro funzioni: AbleICR, AddICRVector, RemICRVector e SetICR.

Queste funzioni sono state aggiunte al sistema con la versione 1.2 per permettere ai programmi di gestire i singoli bit del registro di controllo degli interrupt dei CIA in maniera coerente con la filosofia multitasking. In altre parole, tali funzioni permettono a tutti i task di utilizzare in modo condiviso le risorse CIA, manipolandone i registri e i bit. Possono essere descritte come funzioni per la gestione dei registri dei CIA dedicati al controllo degli interrupt, e inoltre per la creazione e il mantenimento delle routine di gestione di questi interrupt. L'uso di queste funzioni assicura che ciascun task acceda alle risorse dei CIA in maniera appropriata, assicurando la cooperazione fra tutti i task che nel sistema fanno uso di questi chip.

Infatti i dispositivi CIA devono essere considerati "risorse" del sistema, a differenza per esempio del dispositivo Timer, poiché solo un task alla volta può detenere il controllo di un loro bit di registro. L'uso corretto di queste funzioni serve proprio ad assicurare questa condizione.

Poiché queste funzioni possono gestire separatamente i singoli bit dei registri, è possibile che un task stia facendo uso del bit del registro ICR/MASK relativo al Timer-A e un altro stia invece contemporaneamente usando il bit del Timer-B. Non vi può però essere uso simultaneo dello stesso bit, e ciascun task deve "restituire" il bit di cui deteneva il controllo prima che un altro possa farne uso.

Le quattro funzioni sono così definite.

SetICR – permette a un task di forzare la generazione di un interrupt, di disabilitarlo (cioè azzerarne il relativo bit nel registro di controllo), o di avere informazioni sullo stato degli interrupt. Forzare un interrupt significa renderlo immediatamente attivo, operazione che generalmente serve per il debug. Disabilitarlo significa fare in modo che l'interrupt non venga generato anche se si verifica il corrispondente evento. Il controllo infine permette di sapere quali interrupt sono abilitati, cioè sensibili al verificarsi dei relativi eventi.

AbleICR – permette a un task di abilitare o disabilitare gli interrupt relativi ai CIA. Dal momento che un interrupt dev'essere abilitato per far sì che il sistema lo riconosca, gli interrupt disabilitati vengono ignorati e non causano alcuna attività nel sistema. Quando i relativi eventi si verificano, come per esempio l'azzeramento del contatore in uno dei due timer, vengono del tutto ignorati.

AddICRVector – consente di associare una specifica routine di gestione a un bit del registro di controllo degli interrupt di un CIA. Questa è la

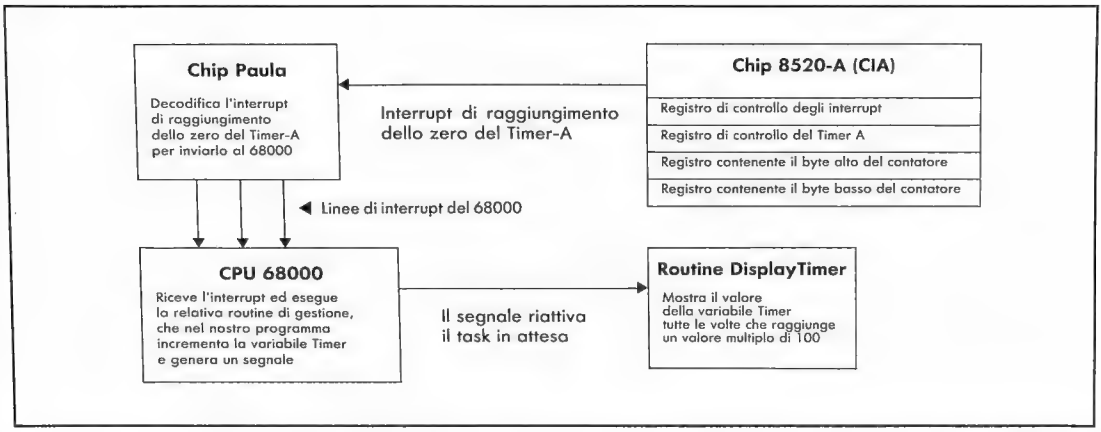


Figura 2: come un interrupt del Timer-A attiva una routine di gestione che generando un segnale dell'Exec "risveglia" un task

routine che viene eseguita quando si verifica l'interrupt. Si può per esempio progettarla perché produca un'animazione grafica con un dato interrupt, e per far questo basta indicarne l'indirizzo nel parametro is_Code di una struttura Interrupt, come viene mostrato nel nostro programma dimostrativo.

RemICRVector – al contrario della precedente, questa routine *rimuove* una data routine di gestione, dissociandola dal bit di interrupt al quale era stata correlata. La routine non viene disallocata, ma quando l'interrupt si verifica nuovamente essa non riceve il controllo. Va usata quando non si desidera più che all'interrupt faccia seguito l'azione precedentemente impostata.

È importante notare che queste funzioni accedono solo ai singoli bit del registro di controllo dei CIA – il registro ICR/MASK di Figura 1 – e non alterano nessun altro registro. In particolare, queste funzioni non modificano il valore dei bit nel registro di controllo A del Timer-A né nel registro CRA di Figura 1 (nel nostro programma alteriamo questi bit in altro modo, come si può notare analizzando le funzioni StartTimer e StopTimer).

Tutte queste funzioni richiedono sempre come primo argomento l'indirizzo della struttura Library che gestisce la risorsa CIA. Nel programma questo indirizzo viene memorizzato all'interno della variabile CIAResource.

Al pari delle librerie shared, come graphics.library, e dei dispositivi, come serial.device, la libreria di gestione della risorsa CIA dev'essere aperta prima che si possa impiegare: chiamando la funzione OpenResource dell'Exec si ottiene l'indirizzo della struttura Library che gestisce la specifica risorsa indicata come argomento. Per i nostri scopi, indichiamo nella chiamata la costante CIAANAME, definita nel file INCLUDE resources/cia.h, che corrisponde alla stringa "ciaa.resource". Il nostro programma mostra come aprire questa libreria, come comportarsi se non si riesce ad aprirla, e come chiuderla.

Discuteremo ora le quattro funzioni nell'ordine in cui vanno normalmente usate in un programma: AddICRVector, SetICR, AbleICR e RemICRVector. Per ognuna, indicheremo come primo argomento la variabile CIAResource.

La funzione AddICRVector. La prima funzione, il cui nome significa "Aggiungi un vettore di gestione di un interrupt", associa la routine di interrupt indicata come terzo argomento al bit del registro ICR/MASK indicato come secondo argomento; l'interrupt relativo a quel bit dev'essere successivamente abilitato con una chiamata alla funzione AbleICR. Con questa operazione, il task si appropria di quell'interrupt, nel senso che fino a quando non chiama RemICRVector per rimuovere la propria routine di interrupt, nessun altro task potrà agganciare a quell'interrupt un'altra routine; in questo modo, AddICRVector e RemICRVector garantiscono una condivisione "esclusiva" di questi interrupt, cioè in tempi diversi.

Gli argomenti di questa funzione sono: *bitNumber*, che indica il numero del bit che si desidera associare alla routine (si noti che non si tratta di una maschera di bit in quanto una stessa routine di interrupt non può essere associata a più interrupt contemporaneamente); *interrupt*, che dev'essere l'indirizzo di una struttura di tipo Interrupt opportunamente allocata e inizializzata dal task che effettua la chiamata. Questa struttura deve indicare l'indirizzo d'inizio della routine di gestione dell'interrupt e l'indirizzo dei relativi dati.

La funzione AddICRVector si usa di solito così:

```
oldInterrupt = AddICRVector (CIAResource, bitNumber,
                             interrupt);
```

La funzione restituisce uno zero nella variabile oldInterrupt se nessun'altra routine di servizio risulta assegnata a quel bit di interrupt; altrimenti restituisce l'indirizzo della routine associata (que-

sta seconda condizione indica il fallimento dell'operazione). Osservando la funzione BeginTimer del nostro programma si vede come è possibile tener conto di questa eventualità.

Per esempio, per collegare una routine di gestione all'interrupt del Timer-A – il bit numero 0 nel registro ICR/MASK – si scrive questa linea di programma:

```
oldInterrupt=AddICRVector(CIAResource,0,interrupt);
```

In tal modo si assegna la struttura Interrupt individuata dall'argomento interrupt al bit numero 0 del registro di controllo, il bit del Timer-A. Ora, dopo che gli interrupt del Timer-A sono stati abilitati con la funzione AbleICR, tutte le volte che questo timer raggiunge lo zero viene mandata in esecuzione la routine specificata nel parametro is_Code della struttura Interrupt indicata come terzo argomento. Nel listato che accompagna l'articolo, all'interno della funzione BeginTimer, questo parametro viene aggiornato con l'indirizzo della routine TimeOut con la seguente istruzione:

```
TimerInterrupt.is_Code = TimeOut;
```

In questo caso TimerInterrupt è il nome della struttura Interrupt che viene associata all'interrupt generato dal conto alla rovescia del Timer-A.

La routine di gestione dell'interrupt, TimeOut, è scritta in Assembly e svolge due compiti: incrementa la variabile Timer e chiama la funzione Signal dell'Exec per risvegliare con un segnale il task in attesa indicandogli che si è verificato un interrupt. Nel nostro programma questo task è la funzione main, la quale all'interno di un ciclo entra in attesa del segnale tramite la funzione Wait. Con questa procedura scavalchiamo la gestione automatica dei segnali condotta dall'Exec, mostrando come può avvenire lo scambio di segnali fra i task.

Per rimuovere la routine di gestione aggiunta al sistema occorre servirsi della funzione RemICRVector.

La funzione SetICR. La seconda funzione (quella che utilizziamo per azzerare nel registro di controllo degli interrupt il bit relativo all'interrupt del Timer-A, cioè per disabilitarlo prima di riabilitarlo) fornisce un modo per azzerare o controllare lo stato di ciascuno dei bit di interrupt presenti nel registro di controllo ICR/MASK, e forzare la generazione di un particolare interrupt. Generalmente SetICR viene impiegata per disabilitare uno o più interrupt appena prima che vengano abilitati (funzione AbleICR), al fine di assicurarsi che la prima chiamata di interrupt non sia causata da fattori non previsti; nel nostro caso, per esempio, si vuole essere sicuri che il primo interrupt sia effettivamente generato dall'azzeramento del contatore nel Timer-A.

SetICR prevede due argomenti: il primo è l'indirizzo base della risorsa, mentre il secondo è una maschera, ossia un insieme di bit a 0 o a 1 corrispondenti ad appropriati bit del registro ICR/MASK, che serve a indicare su quali bit dovrà avere effetto la funzione SetICR.

Se il bit 7 (SET/CLEAR) di questa maschera è a 0, tutti gli altri bit a 1 indicano quali bit nel registro andranno impostati a 0. Se il bit 7 è a 1, gli altri bit a 1 indicano quali bit nel registro andranno impostati a 1, cioè quali interrupt dovranno essere generati artificialmente; quest'ultima operazione serve in genere per il debug. Gli altri bit di questa maschera servono per indicare a quali interrupt ci si riferisce.

Per esempio, per disabilitare l'interrupt del Timer-A del chip 8520-A, si agisce così:

```
SetICR (CIAResource, 0x01);
```

Qui il numero esadecimale 0x01 è equivalente al binario 00000001, in cui il bit 7 a zero indica che vogliamo azzerare alcuni interrupt e il bit 0 a 1 indica che intendiamo agire sul bit di controllo dell'interrupt del Timer-A. Gli altri bit non vengono alterati. Nel listato, all'interno della funzione StartTimer si fa uso delle costanti indicate nel file INCLUDE hardware/cia.h riguardanti i nomi dei bit (per esempio `CLEAR|CIAICRF_TA` nella chiamata a AbleICR).

Per leggere invece il valore del registro ICR/MASK, si utilizza la stessa funzione con un argomento fittizio pari a 0x00:

```
oldMask = SetICR (CIAResource, 0x00);
```

Il task può successivamente analizzare la variabile oldMask per appurare quali interrupt del CIA sono abilitati e quali sono invece disabilitati (bit rispettivamente a 1 e a 0). La maschera di bit così ottenuta riporta lo stato del registro ICR/MASK subito prima che il task ne cambi il contenuto (con il secondo parametro a 0x00 non avviene nessun cambiamento); questa possibilità non è usata nel nostro programma.

La funzione AbleICR. La terza funzione (quella che utilizziamo per abilitare e disabilitare il nostro interrupt) serve ad abilitare e disabilitare gli interrupt del CIA, cioè a indicare al CIA quali eventi devono causare un interrupt (per esempio, l'azzeramento del contatore nel Timer-A) e quali eventi devono invece essere ignorati. Per individuare gli eventi su cui si desidera agire, si fa uso di una maschera di bit il cui contenuto di 0 e 1 si riferisce ai cinque tipi di interrupt del registro di controllo del CIA; anche questa funzione riceve due argomenti e ancora una volta il secondo argomento non è un numero ma una maschera di bit. Per i singoli bit della maschera vale la descrizione fatta per la funzione SetICR.

Al solito, se il bit 7 della maschera è a 0, tutti gli altri bit a 1 indicano quali interrupt devono essere disabilitati, mentre se il bit 7 è a 1 vengono abilitati tutti gli interrupt il cui bit corrispondente è a 1. Proprio come SetICR, questa funzione restituisce il valore precedentemente contenuto nel registro ICR/MASK, che un task può utilizzare per stabilire

quali interrupt sono attivi e quali no, prima di effettuare una seconda chiamata alla funzione AbleICR.

Si noti che nelle librerie ciaa.resource e ciab.resource di sistema le funzioni SetICR e AbleICR sono state progettate per restituire un valore intero, ma dal momento che nel nostro programma non ne teniamo conto, dichiariamo che restituiscono valori di tipo void, cioè che non restituiscono valori significativi.

Esempi d'uso della funzione AbleICR potrebbero essere i seguenti. Per individuare il valore della maschera di bit nel registro di abilitazione degli interrupt si usa la funzione passandole un parametro nullo:

```
oldMask = AbleICR (CIAResource, 0x00);
```

Il task può così esaminare questo valore e intraprendere le azioni opportune. Per abilitare il Timer-A:

```
AbleICR (CIAResource, 0x81);
```

In questo caso, essendo impostato l'ottavo bit, si ha l'effetto di abilitare l'interrupt relativo al bit 0. Per ottenere l'effetto opposto, si fa così:

```
AbleICR (CIAResource, 0x01);
```

Essendo il bit 7 impostato a 0, vengono disabilitati tutti gli interrupt il cui bit corrispondente si trova a 1 (in questo caso solo il bit relativo al Timer-A).

A proposito di questa funzione, chiariamo ora quello che abbiamo anticipato descrivendo SetICR. Nel nostro programma, all'interno della funzione StartTimer, si può notare che prima di abilitare l'interrupt del timer andiamo a disabilitarlo tramite la funzione SetICR. Questa azione "preventiva" è giustificata da una proprietà della funzione AbleICR: se l'interrupt (o gli interrupt) sul quale si agisce è già abilitato, tale interrupt si verifica immediatamente quando la funzione conclude la sua esecuzione, anche se non si è verificato l'evento hardware che dovrebbe esserne la vera causa. Nel nostro caso, la routine di interrupt riceverebbe il controllo non appena AbleICR si completa, cioè in corrispondenza di un interrupt forzato, e non dello scadere del conteggio. Per evitarlo, si chiama appunto SetICR al fine di disabilitare tutti gli interrupt del CIA che intendiamo poi abilitare con AbleICR.

La funzione RemICRVector. L'ultima funzione ("Rimuovi un vettore di interrupt", cioè una routine di gestione di un interrupt) serve a scollegare una data routine di interrupt da un bit del registro ICR/MASK (la routine dev'essere stata assegnata in precedenza con la funzione AddICRVector). I tre argomenti di questa funzione hanno lo stesso significato che avevano con AddICRVector. Non viene restituito alcun valore e una chiamata tipica a questa funzione ha la forma seguente:

```
RemICRVector (CIAResource, BitNumber, interrupt);
```

L'argomento interrupt dev'essere l'indirizzo della stessa struttura Interrupt indicata precedentemente nella chiamata a AddICRVector: la chiamata a RemICRVector dice semplicemente al sistema che quella struttura interrupt non è più associata al bit il cui numero è specificato come secondo parametro. Prendendo come esempio il nostro programma, quando il Timer-A raggiunge nuovamente lo zero dopo l'esecuzione di RemICRVector, la nostra routine di interrupt non riceverà più il controllo fino a quando non si rieseguirà AddICRVector per renderla di nuovo attiva.

Eseguendo RemICRVector, il bit indicato viene lasciato libero per un altro task, che potrà associare all'interrupt relativo una propria struttura Interrupt e quindi una propria routine di gestione, com'è già stato illustrato.

Il programma di esempio

Il listato che accompagna l'articolo si basa su un programma scritto da Paul Higginbottom del Commodore Amiga Technical Support e successivamente pubblicato per facilitare gli esperimenti dei programmatori. Il suo flusso è descritto nella Figura 3 (a pagina 119).

Nonostante si tratti di un semplice nucleo su cui costruire eventualmente programmi più completi, i concetti e i dettagli tecnici sono sufficientemente complessi da farci capire che un programma più complicato avrebbe introdotto troppe idee estranee al nostro soggetto. In particolare, l'output del programma è volutamente scarno e ridotto all'esame del solo valore assunto dalla variabile Timer per il conteggio degli interrupt.

Il programma azzera la variabile Timer e chiama la funzione BeginTimer, la quale cerca di ottenere tutte le risorse necessarie, chiama SetTimer per inizializzare i registri del Timer-A del CIA 8520-A con il numero dal quale deve iniziare il conto alla rovescia (questo numero stabilisce in pratica la frequenza delle chiamate di interrupt), e chiama la funzione StartTimer per attivare il conteggio e per abilitare il relativo interrupt. D'ora in poi la nostra routine di interrupt riceve il controllo ogni volta che il conto alla rovescia raggiunge lo zero. A questo punto il corpo principale del programma entra in attesa di un segnale software attraverso la funzione Wait.

La nostra routine di interrupt, TimeOut, è interamente in Assembly, e ogni volta che il contatore raggiunge lo zero svolge le seguenti operazioni: incrementa la variabile Timer e invia un segnale alla parte principale del programma che ne è in attesa. Questo "si sveglia" (la funzione Wait gli restituisce il controllo) e chiama la funzione DisplayTimer per mostrare a video il valore raggiunto dalla variabile Timer (lo visualizza soltanto quando è un multiplo di cento), e infine chiama di nuovo la funzione Wait in un ciclo che dura finché la variabile Timer non raggiunge il valore 1000.

Il programma consiste di una funzione principale e di altre sei funzioni di contorno: StartTimer,

StopTimer, SetTimer, BeginTimer, EndTimer e DisplayTimer, oltre alla routine di interrupt in Assembly TimeOut (si noti che nel listato compare un'altra parte in Assembly, quella che aggiorna il parametro is_Data della struttura Interrupt). Le parti in Assembly sono delimitate dalle direttive #asm e #endasm. Le funzioni in C fanno a volte uso di funzioni di libreria dell'Amiga per svolgere i loro compiti: tra queste vi sono le quattro funzioni dedicate di cui si è parlato prima e la funzione FindTask.

Il programma è stato compilato con il pacchetto Manx versione 3.6a. Si ricordi che questo compilatore consente l'inserimento nel listato di routine in Assembly, a differenza del Lattice. Se si possiede il Lattice occorre quindi compilare TimeOut separatamente e sottoporla poi a link insieme al corpo principale del programma in C.

Il programma, come già detto, non prevede alcuna interazione con l'utente e si limita a produrre una serie di eventi temporizzati riportando sul video i valori assunti di volta in volta dalla variabile Timer. Si può comunque modificare il valore del parametro TIME_SLICE prima della compilazione, per variare gli intervalli di tempo tra due interrupt: TIME-SLICE rappresenta la durata del conto alla rovescia espresso in microsecondi.

In alternativa, si può modificare il programma principale affinché richieda all'utente di fornire il parametro TIME_SLICE durante l'esecuzione, ricorrendo alla funzione scanf del C. Quando il valore di questo parametro è noto, il programma va a vedere se la funzione BeginTimer restituisce un valore diverso da zero. La funzione BeginTimer può tornare in quattro modi diversi al programma che ha effettuato la chiamata, a seconda dell'esito della sua ricerca delle risorse di sistema: restituisce un valore 1 (TRUE) solo se si verificano le seguenti condizioni.

– È stato possibile trovare e allocare un bit di segnale utilizzando la funzione AllocSignal, ossia tale funzione ha restituito un numero di bit di segnale diverso da zero nella variabile TimerSigBit del programma. Dal momento che l'argomento è –1 (qualsiasi bit di segnale va bene), perché venga restituito un valore diverso da 0 basta che fra i 16 possibili bit di segnale disponibili per il task ne esista uno non ancora allocato.
– La funzione è riuscita ad aprire la risorsa CIA utilizzando la funzione OpenResource, e quindi il puntatore CIAResource ha assunto un valore diverso da zero: ciò significa che la libreria della risorsa CIA è stata trovata e aperta correttamente.
– La funzione è riuscita a installare la struttura Interrupt di nome TimerInterrupt nel sistema facendo uso della funzione AddICRVector, cioè il valore restituito da questa funzione era nullo. Se invece viene restituito un valore diverso da zero significa che l'interrupt indicato è attualmente impiegato da un altro task. Ancora una volta è importante notare che, al contrario di quanto avviene per l'allocazione dei bit di segnale dove abbiamo 16 possibilità, qui c'è un solo bit che riguarda il Timer-A del CIA 8520-A, ed è quindi richiesta un'aderenza assoluta al protocollo di condivisione del registro ICR/MASK tra i vari processi.

Se tutte queste condizioni sono verificate, BeginTimer prosegue la propria esecuzione senza fermarsi e chiamando le funzioni SetTimer e StartTimer; se invece anche una sola delle precedenti condizioni non si verifica, restituisce un valore 0 (FALSE) costringendo il programma principale a eseguire la funzione EndTimer e a terminare l'esecuzione.

Con la chiamata a SetTimer viene impostata la durata del conto alla rovescia del Timer-A, mentre StartTimer attiva il conteggio. Successivamente il programma principale entra in un loop in attesa di un segnale. Questo segnale viene inviato dalla routine di interrupt TimeOut, la quale fa uso dell'apposita routine dell'Exec nell'istruzione `jsr _LVOSignal(a6)`.

Il programma principale viene posto in attesa tramite la funzione dell'Exec Wait, alla quale viene passato come parametro il numero di segnale contenuto nella variabile TimerSigMask (ottenuto in precedenza). Questo è il segnale che viene inviato quando il Timer-A del chip 8520-A arriva a zero generando l'interrupt.

Si noti che la prima operazione svolta dalla funzione BeginTimer è chiamare la funzione FindTask allo scopo di ottenere l'indirizzo del puntatore alla struttura Task che rappresenta il task corrente. Questa informazione serve per comunicare al programma qual è il task destinatario del segnale.

Quando la funzione Wait restituisce il controllo, il programma principale fa partire l'esecuzione della funzione DisplayTimer: questa svolge alcuni semplici calcoli sulla variabile Timer e successivamente crea un output sul video.

Il programma può per comodità essere compilato utilizzando il file script di compilazione e link presentato nello scorso numero di *Commodore Gazette*. Tra i vari flag attivabili nella compilazione, particolare importanza riveste +P, che tiene conto del fatto che due registri della CPU devono essere salvati perché il task possa contenere routine di interrupt. Se abbiamo a disposizione il file sorgente Temporizzazioni.c, la compilazione può essere ottenuta dal CLI con il comando:

```
execute ccln temporizzazioni
```

Questo produce il file eseguibile Timer, che può essere utilizzato scrivendo banalmente "Temporizzazioni" seguito da RETURN. Dopo un certo ritardo, dovuto alla fase d'impostazione del timer, vengono visualizzati uno dopo l'altro i valori multipli di 100 assunti dalla variabile Timer. Questa variabile viene incrementata tutte le volte

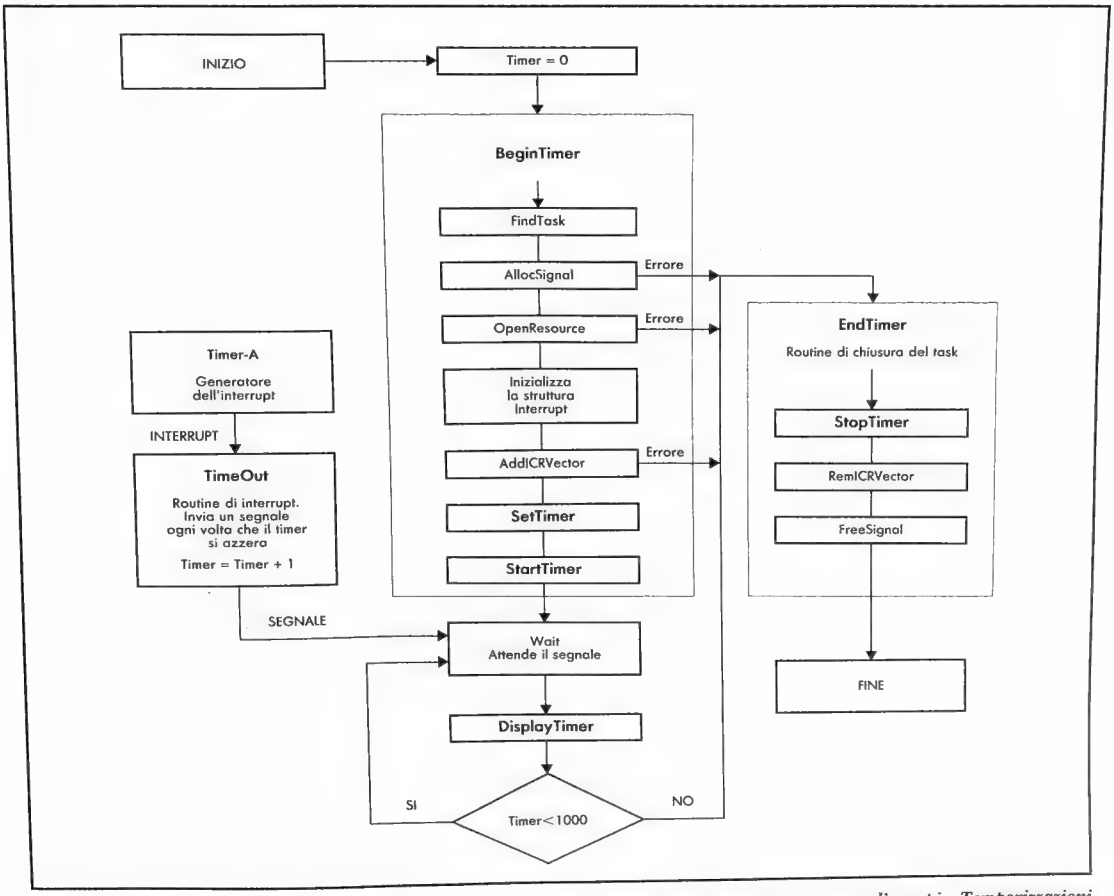


Figura 3: diagramma di flusso relativo a tutte le operazioni effettuate dal nostro programma d'esempio Temporizzazioni

che Timer-A genera un interrupt.

A ognuno di questi interrupt, la relativa routine di gestione oltre a incrementare Timer invia anche un segnale al task in attesa, il quale chiama la funzione DisplayTimer. Ne viene però mostrato il valore soltanto ogni 100 interrupt. Il numero d'interlinee che appare tra questi numeri dipende dal ciclo for della funzione DisplayTimer e aumenta proporzionalmente con il numero di volte che la funzione visualizza un valore di Timer.

Ovviamente, si può anche fare in modo che la routine di interrupt TimeOut invii un segnale al task ogni 100 volte che viene eseguita, cioè ogni 100 chiamate di interrupt.

È importante notare che le funzioni SetTimer, BeginTimer e StartTimer sono richiamate soltanto una volta durante l'esecuzione del programma. Tuttavia, dopo la prima impostazione, il timer viene reimpostato automaticamente, perché nella funzione StartTimer viene impostato il flag CIACRAF_RUNMODE del parametro ciacr.

Il ciclo più interno si ferma soltanto quando la variabile Timer raggiunge quota 1000: a quel punto il programma principale esce dal loop che abbiamo descritto e chiama la funzione EndTimer per ripristinare le condizioni iniziali e concludere finalmente la sua esecuzione.

La prossima puntata

Analizzando questo programma si scopre presto che è necessario un ulteriore approfondimento. Benché sia piuttosto breve, infatti, presenta alcune complessità che meritano una spiegazione più accurata. Per questo motivo il nostro prossimo articolo proseguirà a livello più profondo l'analisi di questo esempio e di altri a esso collegati. Di ciascuna funzione esamineremo la ragion d'essere e il modo in cui svolge il proprio compito.

Il secondo obiettivo del prossimo articolo sarà l'esame dei programmi dal punto di vista di chi vuole scrivere generico software per l'Amiga interfacciandosi direttamente con i registri hardware. Vedremo come viene affrontato e superato questo problema, e come vengono creati i vari blocchi.

Listato: *Temporizzazioni*

```
#include <exec/types.h>
#include <exec/tasks.h>
#include <exec/interrupts.h>
#include <hardware/custom.h>
#include <hardware/intbits.h>
#include <hardware/cia.h>
#include <resources/cia.h>

int                 TimerBitSegn = -1;
long                TimerMascSegn;
USHORT              Timer;

struct Interrupt    TimerInterrupt;
struct Interrupt    *OldCIAInterrupt =
                          (struct Interrupt *)-1;
struct Library      *CIAResource = NULL;
struct Task         *questoTask;

#define ciatlo      ciaa.ciatalo
#define ciathi      ciaa.ciatahi
#define ciacr       ciaa.ciacra
#define CIAINTBIT   CIAICRB_TA
#define CLEAR       0
#define TIME_SLICE ((USHORT) 10000)

void StartTimer()
{
   void  SetICR(), AbleICR();
   ciacr &= ~CIACRAF_RUNMODE;
   ciacr |= CIACRAF_LOAD | CIACRAF_START;
   SetICR (CIAResource,CLEAR | CIAICRF_TA);
   AbleICR(CIAResource,CIAICRF_SETCLR | CIAICRF_TA);

}


void StopTimer()
{
   void  AbleICR();
   AbleICR (CIAResource, CLEAR | CIAICRF_TA);
   ciacr &= ~CIACRAF_START;
}


void SetTimer(microsec)
USHORT microsec;
{
   ciatlo = microsec & 0xFF;
   ciathi = microsec >> 8;
}


BOOL BeginTimer()
{
   long              AllocSignal();
   void              TimeOut();
   struct Library    *OpenResource();
   struct Interrupt *AddICRVector();
   struct Task       *FindTask();

   questoTask = FindTask(NULL);
   if ((TimerBitSegn = AllocSignal(-1)) == -1)
   {
      puts("Timer: AllocSignal e' fallita");
      return(FALSE);
   }
   TimerMascSegn = 1L << TimerBitSegn;
   if (!(CIAResource = OpenResource(CIAANAME)))
   {
      printf("Timer: non riesco ad aprire %s",
             CIAANAME);
      return(FALSE);
   }
   TimerInterrupt.is_Node.ln_Type = NT_INTERRUPT;
   TimerInterrupt.is_Node.ln_Pri  = 127;

#asm
   include "exec/types.i"
   include "exec/interrupts.i"

   lea    _TimerInterrupt,A0
   move.l A4,IS_DATA(A0)
#endasm

   TimerInterrupt.is_Code = TimeOut;
   if (OldCIAInterrupt = AddICRVector(CIAResource,
       CIAINTBIT, &TimerInterrupt))
   {
      puts("Timer: l'interrupt e' gia' in uso");
      return(FALSE);
   }
   SetTimer(TIME_SLICE);
   StartTimer();
   return(TRUE);
}

void EndTimer()
{
   if (TimerBitSegn != -1)
   {
      if (!(OldCIAInterrupt))
      {
          StopTimer();
          RemICRVector(CIAResource, CIAINTBIT,
                       &TimerInterrupt);
      }
      FreeSignal(TimerBitSegn);
   }
}

#asm
   public _Timer,_LVOSignal,_questoTask
   public _TimeOut

_TimeOut:
   move.l A4,A5
   move.l A1,A4
   addq.w #1,_Timer
   move.l _TimerMascSegn,D0
   move.l _questoTask,A1
   jsr    _LVOSignal(A6)
   move.l A5,A4
   rts
#endasm

void DisplayTimer()
{
   int    i;
   USHORT TTimer;

   TTimer = Timer;
   if (!(Timer % 100))
   {
      for (i=0; i < TTimer/100; ++i)
      {
         puts(".");
      }
      printf("%d\n", TTimer);
   }
}

main()
{
   Timer = 0;
   if (BeginTimer())
   {
      do
      {
          Wait(TimerMascSegn);
          DisplayTimer();
      }
      while (Timer < 1000);
   }
   EndTimer();
}
```

■

GEOS

NON HA PIÙ SEGRETI

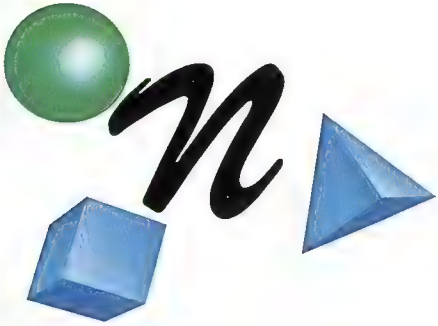
Sei un utente di GEOS e ne sei entusiasta? Desideri scoprirne ogni dettaglio e impararne il funzionamento? Vuoi programmare in questo sistema operativo ad ambiente grafico? Allora questo volume della IHT ti è indispensabile. Scritta dagli stessi creatori di GEOS, questa guida svela tutti i segreti del sistema operativo e permette di creare stupende applicazioni per il C-64 e il C-128 dotate di interfaccia utente grafica, menu, icone, finestre, box di dialogo, fonti proporzionali, processi in multitasking, RAM disk, gestione dei file in overlay, controllo del mouse, driver di input e di stampa, e tante altre caratteristiche che fanno di GEOS un ambiente di lavoro e sviluppo davvero professionale.

592 pagine. L. 64.000

I libri IHT sono disponibili nelle migliori librerie e computer shop. Per ordini diretti servirsi dell'apposito modulo pubblicato a pagina 127
IHT Gruppo Editoriale - Via Monte Napoleone, 9 - 20121 Milano - Tel. 02/794181-794122 - Fax 784021 - Telex 334261 IHT I
Distribuzione: Messaggerie Libri - Via Giulio Carcano, 32 - 20141 Milano - Tel. 02/8438141

COMMODORE NEWS

NOVITÀ HARD E SOFTWARE DALL'ITALIA E DAL MONDO



ITALIA

AIIM '89

Si svolgerà a Como, dal 5 al 7 ottobre 1989, il Congresso Nazionale dell'Associazione Italiana d'Informatica Medica (AIIM), in concomitanza con la Fiera Internazionale del software per la medicina.

Il programma si articola nei seguenti punti: minicorsi; sessioni scientifiche; sessioni di dimostrazioni di prototipi; profili di enti, associazioni, grandi progetti o aziende; minisimposi dedicati ad argomenti rilevanti discussi da ospiti internazionali; stand di espositori; scambio di software.

Centro di Cultura Scientifica "Alessandro Volta"
Villa Olmo, Via S. Cantoni, 1
22100 Como
(Tel. 031/572213)

SMAU E COMPUTER PLAY

Il 9 ottobre prossimo, nel corso dello SMAU, avrà luogo la premiazione del Concorso Computer Play '89 organizzato da SMAU e A.I.C.A. e giunto quest'anno alla settima edizione. Il concorso si rivolge ai giovani fino ai 21 anni e chiede loro di proporre programmi per PC o programmi di gestione hardware che valorizzino un uso "amichevole" del computer.

Durante lo SMAU, la sezione Computer Play disporrà di uno stand dove i giovani potranno proporre i loro lavori o assistere alla dimostrazione dei programmi "concorrenti": uno spazio dedicato espressamente agli incontri e allo scambio di esperienze tra i giovani.

Segreteria organizzativa e ufficio stampa:
Federica Bergamaschi
Studio giornalistico Edit
Via A. Da Recanate, 4
20124 Milano
(Tel. 02/6692681)

SOFTWARE PER IL TERRITORIO

Il prossimo 11 ottobre si terrà, presso la Villa Olmo di Como, la "Seconda Mostra Internazionale di Software per l'Ambiente e il Territorio". Alla manifestazione parteciperanno ricercatori ed enti di vari Paesi europei così da garantire un proficuo scambio di esperienze e di opinioni. L'ingresso alla mostra sarà gratuito e consentirà ai visitatori di sperimentare direttamente su personal computer gran parte del software presentato. Sarà inoltre disponibile un catalogo dettagliato di tutti i programmi.

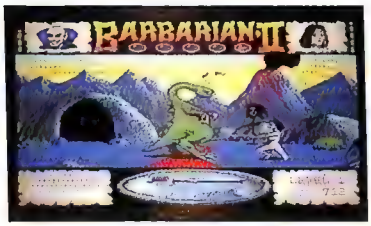
Prof. Giorgio Guariso
Politecnico di Milano (Dip. Elettronica)
Via Ponzio, 35/5
20133 Milano
(Tel. 02/23993560)

ESTERO

BARBARIAN II PER AMIGA

La software house inglese Palace Software annuncia la disponibilità della versione per l'Amiga di *Barbarian II: The Dungeon of Drax*. Nel trasferimento è stata mantenuta tutta la giocabilità originale, e in più sono stati notevolmente migliorati colore e suono. Gli effetti sonori sono analoghi a quelli della versione per l'Atari ST, ma i quattro canali stereo dell'Amiga li fanno sembrare di un altro pianeta.

Attualmente la versione per l'Atari ST ha conquistato la vetta delle classifiche in Francia e in



Inghilterra, e quella per l'Amstrad è al primo posto in Francia.

Palace Software Ltd
The Old Forge Business Centre
7 Caledonian Road
London N1 9DX, England
(Tel. 0044/1/2780751)

MICROPROSE NEWS

Acquisizioni. La MicroProse ha acquistato la Telecomsoft dalla società britannica Telecom, e ora detiene tutti i diritti sui marchi, sui prodotti e sui programmi di sviluppo di questa Casa. I piani attuali prevedono un rilancio dei marchi Firebird e Rainbird, mentre non si è ancora raggiunta nessuna decisione per il marchio Silverbird.

Premi. La versione PC del simulatore di combattimenti aerei *F-19 Stealth Fighter* della MicroProse ha vinto il prestigioso premio assegnato ogni anno dalla Software Publishers Association, nella categoria "Best Simulation". La versione per C-64/128 del gioco è stata recensita in *Commodore Gazette* n. 2/88.

Nuove tecnologie. La Microprose Games, sussidiaria della MicroProse Software per il settore coin op, entra prepotentemente nel settore annunciando le sue prime uscite per il prossimo autunno. Promette prodotti all'avanguardia che richiederanno hardware totalmente nuovo: grafica 3D con effetti "multidimensionali" dovuti alla generazione di 60 mila poligoni al secondo (il limite attuale è di circa 12 mila), e uno schermo che si rigenera 30 volte al secondo. Benché il gioco in progetto non abbia ancora un nome, sarà molto probabilmente un simulatore nato sull'onda del successo di *Stealth Fighter*.

MicroProse Software Ltd
Unit 1, Hampton Street, Tetbury
Glos. GL8 8DA, England
(Tel. 0044/666/54326)

ANIM... AEGIS

L'ultima novità della Aegis nel settore dell'animazione e degli effetti speciali è *ANIMagic* per l'Amiga, un programma che lavora con *Videoscape 3D*, *Deluxe Paint III* e con tutti i più famosi programmi di disegno e d'animazione che producono file ANIM o IFF. Questo programma è un editor con cui è possibile modificare o fondere fra loro animazioni e disegni, ottenendo spettacolari effetti speciali... pur di avere almeno un mega di RAM.

Aegis Development Inc.
2115, Pico Blvd.
Santa Monica, CA 90405, USA
(Tel. 001/213/3929972)

ELECTRONIC ARTS NEWS

La Electronic Arts annuncia la prossima uscita di *The Hound of Shadow* (Amiga), un gioco con una tematica da brivido. Si inserisce tra i lovecraftiani Miti di Cthulhu, e si avvale di un nuovo sistema per giochi di ruolo computerizzati, il Timeline.

Il gioco è ambientato negli anni Venti, e sono state fatte accurate ricerche per assicurare una perfetta ricostruzione storica. Il giocatore riveste il ruolo di un investigatore che rivela gradualmente arcani misteri e porta alla luce gli innominabili orrori che si celano dietro il volto apparentemente tranquillo di una normale cittadina inglese del 1920.

Il Timeline rende possibile utilizzare lo stesso personaggio in diverse avventure, mantenendo l'abilità e l'esperienza raggiunta fino a quel momento.

Electronic Arts
11-49 Station Road, Langley
Berkshire SL3 8YN, England
(Tel. 0044/753/49442) ■

Manoscritti: le collaborazioni dei lettori — manoscritti, disegni e/o fotografie — sono benvenute e verranno valutate in vista di una possibile pubblicazione. Commodore Gazette non si assume comunque responsabilità per perdite o danni al materiale. Si prega di allegare una busta affrancata e indirizzata per ogni articolo. Il pagamento per materiale non richiesto viene effettuato solo in seguito all'accettazione da parte della redazione. I contributi editoriali (di qualunque forma) non si restituiscono. Tutta la corrispondenza editoriale, richieste di annunci, problemi di sottoscrizione abbonamenti, di diffusione e con gli inserzionisti, deve essere indirizzata a: Commodore Gazette — Uffici Editoriali — Via Monte Napoleone, 9 — 20121 Milano
Commodore Gazette è un periodico indipendente non connesso in alcun modo con la Commodore Business Machines e con tutte le sue sussidiarie e affiliate, compresa la Commodore Italiana S.p.A. Commodore Gazette viene pubblicata mensilmente dalla IHT Gruppo Editoriale, Via Monte Napoleone 9, 20121 Milano. Il costo degli abbonamenti è il seguente: Italia: 12 numeri L. 96.000, 24 numeri L. 192.000, 36 numeri L. 288.000. Estero: Europa L. 150.000 (10 numeri), Americhe, Asia... L. 200.000 (10 numeri). Nessuna parte di questa pubblicazione può essere in alcun modo riprodotta senza il permesso scritto dell'editore. La redazione si adopera per fornire la massima accuratezza negli articoli e nei listati pubblicati. Commodore Gazette non si assume responsabilità per eventuali danni dovuti a errori od omissioni.

Classified

Software

Scambio collezione completa Fish Disk (1 – 210) per Amiga con altro software di pubblico dominio su disco. Inviare lista o telefonare ore serali. Luigi Callegari, Casella Postale 15 – 21040 Sumirago (VA) – Tel. 0331/909183.

Scambio software per Amiga e Commodore 64. Contattatemi: Leonardo Pesco – Via S. Nicastro, 4 – 91026 Mazara (TP) – Tel. 0923/945623.

Scambio/compro programmi per Commodore Amiga. Annuncio sempre valido. Per informazioni telefonare allo 0332/212822 (Varese) e chiedere di Fabio.

Scambio programmi per Amiga preferibilmente giochi di ruolo, in tutta Italia! Telefonare o scrivere a Fabrizio Fantoni – Via Dal Molin 51/A – 25015 Desenzano (BS) – Tel. 030/9140144.

Programmi Amiga scrivere o telefonare a Danilo Cornia, Via Inzani, 1 – 29100 Piacenza – Tel. 0523/65756.

Attenzione! Vendo in blocco 50 dischi pieni di programmi per il "glorioso" C-64, al prezzo stracciato di lire 60.000 (con disco). Affrettatevi: per ogni informazione telefonare allo 0771/741263 e chiedere di Luigi. Preferibilmente provincia di Latina.

Vendo circa 400 programmi per C-64/128 a L. 130.000 non trattabili, causa passaggio ad Amiga. Scambio anche programmi per Amiga. Scrivere a Francesco Cuzzola – Via Rucellai 20/3 – 20126 Milano.

Scambio software per Amiga nella zona di Latina. Scrivete a Paolo Giannetti – Viale Della Vittoria, 8 – 04019 Terracina (LT).

Vendo i seguenti programmi originali: *Stealth Fighter*, *Night Rider*, *Dark Side*, *SDI Sega*, *B-24 Simulator*, *Bismark* su cassetta a Lire 10.000 cadauno. *Questron II*, *Roadwar 2000*, *Echelon* su disco a Lire 18.000 cadauno spese postali escluse. In blocco Lire 90.000 spese incluse. Francesco di Maio – Viale Nettuno, 152 – 66023 Francavilla (CH) – Tel. 085/817520, ore pasti.

Scambio programmi per Amiga. Annuncio sempre valido. Scrivere a Massimo Bianco – Via Lucania, 65 – 74100 Taranto.

Programmi per Commodore 64 telefonami o richiedi l'ultima lista: Fabrizio Lolli – Via Sarfatti, 26/4 – 20136 Milano – Tel. 02/8375450.

Tutto per Amiga ai prezzi più bassi d'Italia. Richiedere catalogo gratuito. Vladimiro Jovanovic – Via Trieste, 18 – 65010 Cerratina, (PE) – Tel. 085/9771045.

Scambio programmi per C-64/128 e CP/M. Annuncio sempre valido. Inviare le vostre liste a: Lorenzo Lascialfari – Via Donizetti, 79 – 50018 Scandicci (FI) – Tel. 055/753189.

Scambio programmi per il favoloso computer Amiga. Telefonatemi o scrivetemi, rispondo a chiunque. Mauro Bricca – Via Monade, 38 – 18013 Diano Marina (IM) – Tel. 0183/400814-495491.

Scambio programmi per C-128, sia in modo 128 che in CP/M. Annuncio sempre valido. Cerco inoltre un foradischetti in buone condizioni. Inviare vostre liste a: Daniele Poletti – Via Cavo Benedettino, 11 – 44048 S. Maria Codifiume (FE).

Vendo 50 programmi originali su disco per C-64 a L. 8.000 l'uno. Valter Rivoira – Tel. 0175/42529 Saluzzo (CN).

Vendo in blocco 50 dischi pieni di programmi per il C-64 a L. 50.000 (comprese eventuali spese di spedizione). Inoltre compro i seguenti programmi: *War in Middle Earth*, *Blasteroids*, *R-Type*, *Denaris*, *Last Duel* per C-64 su disco, completi di manuali. Per maggiori informazioni scrivere a: Massimo Tabasso – Piazza Molineris, 1 – 12038 Savigliano (CN).

CLASSIFIED DELLA COMMODORE GAZETTE È UN MODO ECONOMICO PER INFORMARE LA PIÙ VASTA UTENZA COMMODORE SUI VOSTRI PRODOTTI O SERVIZI. GLI ANNUNCI NON A SCOPO DI LUCRO, INVIATI DA PRIVATI, VENGONO PUBBLICATI GRATUITAMENTE (COMPILARE L'APPOSITA SCHEDA DI SERVIZIO LETTORI).

Quote: 15.000 lire per linea, minimo 4 linee. Aggiungere 5.000 lire per ogni parola in grassetto o 50.000 lire per l'intero annuncio in grassetto.
Condizioni: pagamento anticipato. Vengono accettati assegni e vaglia postali. Gli assegni devono essere intestati a: IHT GRUPPO EDITORIALE s.r.l.
Forma: gli annunci sono soggetti all'approvazione dell'editore e devono essere scritti a macchina o in modo molto chiaro. Una linea equivale a 40 lettere, spazi tra le parole compresi. Pregasi sottolineare le parole che si intendono scrivere in grassetto.
Informazioni generali: gli inserzionisti devono sempre specificare nome e indirizzo completo. Gli annunci appariranno nel primo numero disponibile dopo il ricevimento.

Inviare il materiale a:
IHT GRUPPO EDITORIALE
UFFICI PUBBLICITARI
VIA MONTE NAPOLEONE, 9
20121 MILANO

Attenzione: Commodore Gazette non si assume responsabilità in caso di reclami di qualunque natura da parte degli inserzionisti e/o dei lettori. Nessuna responsabilità è altresì accettata per errori e/o omissioni di qualsiasi tipo.

Acquisto e scambio programmi per Amiga 500. Inviare le vostre liste, assicuro risposta. Annuncio sempre valido. Cristiano Cravanzola – Via Professor Oliva, 16 – 12011 Borgo San Dalmazzo (CN).

Scambio programmi per Commodore 128, sia in modo 128 che in CP/M. Vasta biblioteca software compreso GEOS 128 2.0. Annuncio sempre valido. Inviare le vostre liste a: Guido Baralla – C. P. 20 – 55040 Ripa (LU) – Tel. 0584/760693.

Scambio programmi per C-128 sia in modo 128 che in CP/M. Annuncio sempre valido. Inviare le vostre liste a: Rosario Albanese – Corso Alberto Amedeo, 66 – 90138 Palermo – Tel. 091/327764.

Tutte le novità d'importazione per Amiga e per Commodore 64. Programmi originali, funzionanti garantiti. Scrivere a: Giuseppe Borracci – Via Mameli, 15 – 33100 Udine – Tel. 0432/580157.

Vendo sistema completo GEOS V1.3 con *geoCalc*, *geoFile*, *Font pack 1*, *Desk pack 1*, *Writer's workshop*, *geoPublish*, *Font pack 2*, *Q-link V4.0*, *geoDex*, *GEOS upgrade V1.3* e *Geos Busters* + manuale in inglese e guida alla programmazione di GEOS in italiano. Telefonare allo 010/620209 (Genova).

Per Amiga scambio programmi; posseggo vasto archivio con continui aggiornamenti. Scrivere o telefonare a: Antonio Civitelli – Viale Gramsci, 130 – 52027 San Giovanni Valdarno (Arezzo) – Tel. 055/940342 dalle ore 19 alle ore 21.30. Massima serietà.

Vendo vastissimo assortimento di programmi in MS-DOS completi di manuale d'uso, su dischi da 3,5'' o 5,25''. Ingegneria, Cad-Cam, Desktop, grafica, elettronica, utility, gestionali ecc. tutti a prezzi modici. Per informazioni: Paolo – Tel. 0587/685513-55438.

Il Realtime Group scambia qualsiasi tipo di programma per Amiga. Scrivere o telefonare a: Davide Pedroli – Via Madonna Pellegrina, 125 – 20010 Bareggio (MI) – Tel. 02/9013501.

Scambio programmi per Amiga. Inviate le vostre liste, rispondo a tutti. Modifico inoltre il monitor 1901 per adattarlo all'Amiga. Giuseppe Di Bello – Via dei Negri, 41 – 28047 Oleggio (Novara) – Tel. 0321/93622.

Scambio programmi per Amiga. Contattatemi, rispondo a tutti. Luca Ceria – Via Roma, 46 – 13060 Valdengo (VC).

Compro, scambio software Amiga con software originale tedesco e canadese. Massima serietà e velocità. Inviare le vostre liste a: Gaetano Lanzalone – Piazza Duomo, 4 – 95024 Acireale (Catania).

Scambio programmi per Amiga. Massima serietà. Annuncio sempre valido. Inviare le vostre liste a: Aldo Imbriaco – Via F. Cammarota, 12 – 84078 Vallo della Lucania (Salerno).

Cerco programmi totocalcio tipo *Totoplus, Tot 13* ecc. per C-64, programmi per stampare, programmi grafici, per titolare film. Massima serietà. Inviare le vostre liste a: Luigi Cirillo – Via Ghiaie, 20 – 38100 Trento.

Hardware

Vendo Amiga 500 + drive esterno + monitor a colori con sintonizzatore + 100 dischi utility/giochi a lire 1.900.000. Telefonare allo 0432/580157 ore pasti.

Vendo un C-128, fornito di drive 1571, due joystick, un registratore originale, numerosi programmi e modem 6499. Tutto a L. 750.000. Scrivere o telefonare a Pippo Riccardo – Via Fosson, 22 – 30020 Annone Veneto (VE) – Tel. 0422/769170.

Cerco espansione da 512K per Amiga 500 a prezzo modico. Scrivere o telefonare a: Fabio Fabbi – Via Montecuccoli, 119 – 41010 Limidi (MO) – Tel. 059/561586 ore pasti.

Vendo drive 1541 + portadischi e dischi a L. 230.000 trattabili. Cerco stampante per A500 e Prosound Designer. Nicola Finelli – Via Gesso, 137 – 40069 Zola Predosa (BO).

Vendo causa passaggio a sistema superiore: interfaccia midi C-64; programmi sequencer: *Pro 16 + TNS + Edit kit, C-lab, Jms Siel Live*. Telefonare ore pasti ad Andrea – Tel. 039/674280.

Vendo: C-64, 1541, MPS803, registratore, 4 libri, (uno su GEOS della IHT, due di Assembler), 40 dischi e 70 cassette piene di programmi (circa 800 anche GEOS 1.3 e tra i quali *Fontpack1, geospell, Dragon's Lair, Shoot'em up Constr.* kit originali), circa 20 riviste, cartuccia sprotettrice, 2 schede duplicatrici a sole lire 900.000. Contattare Marco Averone – Via Ribet, 9 – 10125 Torino – Tel. 011/657239.

Cercasi disperatamente drive 1571 in perfette, ripeto perfette condizioni. Possibilmente prezzo basso. Adriano – Tel. 0565/977539 (Isola d'Elba).

Compriamo hardware per Amiga e MS-DOS di qualunque genere solo se in perfetto stato e a ottimo prezzo. Inoltre cediamo software per Amiga e MS-DOS in cambio di programmi per MS-DOS. Diegosoft (DGS) – Viale Cortemaggiore, 12 – 93012 Gela (CL) – Tel. 0933/938404.

Vendo C-128 + disk drive 1571 + MPS 1200 + modem 6499 a L. 1.000.000 trattabili. Per informazioni telefonare o scrivere a: Mario Fazio – Via Roma, 306 – 00011 Guidonia (Roma) – Tel. 0774/378128.

Varie

Vendo Honda 350 Four, 1974, rossa, con borse, gomme e marmitta in buono stato, motore da revisionare. Prezzo da concordare. Luca, ore serali, Tel. 02/8135727.

Per Amiga scambio esperienze (linguaggio C). Per informazioni telefonare a: Valter Mazzola – Tel. 0362/590753 (provincia Milano).

Vendo per Amiga 60 manuali nuovi di utilità e giochi, il tutto a L. 100.000. Scrivere o telefonare a: Tiziano Rossi – Via Stanga, 11 – 37139 Verona – Tel. 045/565012.

Vendo Amiga 1000 tastiera italiana, monitor 1081, mouse, drive esterno, espansione a 1.5 MB, stampante Panasonic 1031 con supporto, 800 programmi su dischi da 3,5 classificati e numerati in 6 contenitori Posso. Digitalizzatori audio, vari accessori e giochi originali. Tutto in blocco a L. 3.200.000. Telefonare ore pasti a: Riccardo – Tel. 055/685216.

Vendo annate di Amiga World a metà prezzo – Marco Sivori – Via Barchetta, 18/9 – 16162 Bolzaneto (GE).

Amighi miei! Non fermiamo la sete di software che i nostri computer hanno! Contact me! Ti aspettano: news, super intro, trucchi, segreti, hardware e altro ancora. Aldo Imbriaco – Via F. Cammarota, 12 – 84078 Vallo della Lucania (SA).

Maui BBS System, nuovo BBS per utenti Commodore. Numero telefonico 0862/66468. Velocità 300/1200/2400 BPS. Orario 20/24 (festivi e prefestivi 18/24). Numerose aree programmi e messaggi. Vi aspetto.

È nata la O.A.Sys BBS, banca dati pescarese ad accesso gratuito. Parametri 085/77717, da 300 a 1200, 8, N, 1. Aree MS-DOS, Amiga, C-64 e 128, Atari, Apple con software di pubblico dominio.

Desidero entrare in rapporto di scambio programmi ed esperienze con possessori di Amiga 2000. Scrivere a: Francesco Cuzzola – Via Rucellai, 20/3 – 20126 Milano – Tel. 02/2550620.

Vendo riviste: MC microcomputer, num. 43, 44 e dal 46 al 53. SuperVic e C-64 num. 2, 4, 5, 6. SuperCommodore dal n. 1 al n. 7. Noi 64 e 128 primi 3 numeri. Le ultime 3 testate sono comprensive di cassette programmi. Telefonare a: Antonio Di Palma – Tel. 081/923810 – Nocera Inf. (SA).

Scambio e vendo programmi per C-64 e 128. Vendo stampante Commodore MPS-801 a L. 220.000 trattabili e corso di programmazione Basic per C-64 della Beatrice d'Este (14 cassette). Scrivere o telefonare a: Marco Massari – Via O. Messori, 46 – 41012 Carpi (MO) – Tel. 059/681120.

Cerco appassionati di robotica applicata ai computer C-64/128/A2000. Matteo Mairate – Via Fiume, 96 – 10088 Volpiano (TO) – Tel. 011/9884461.

Commodore Club

Leosoft Club Amiga e C-64 è veramente diverso. Iscrizione gratuita, software di qualsiasi tipo, scambi, tutte le ultime novità ecc. Scrivi per informazioni a: Leosoft Club – Via Seb. Nicastro, 4 – 91026 Mazara del Vallo (TP) – Tel. 0923/945623-931302.

Amiga Club cerca nuovi soci: abbonarsi non costa niente. Prezzi vantaggiosissimi. Importazione diretta e manualistica in italiano come: *Sculpt 3D, Videoscape, AmigaDOS, Deluxe Video, City Desk, Page Setter, Deluxe Paint II, Butcher*. Se vuoi del software selezionato telefona o scrivi a Ugo Bolzoni – Via S. Bellino, 8 – 45100 Rovigo – Tel. 0425/22619.

Amiga Club Verano scambia programmi per Amiga in provincia di Milano. Telefonare ore serali a Danilo 0362/900057 o Alessandro 0362/901587.

Free Amiga Club. I programmi sono gratis!! Scrivere per informazioni allegando bollo per la risposta. Andrea Focardi – Via G. Di Vittorio, 56 – 50015 Grassina (FI).

Finalmente a Pescara è nato il Crazy Club Amiga. Tutti coloro che volessero far parte di questo club possono telefonare allo 085/690350. Giochi, utility, e hardware a disposizione dei nostri futuri soci.

AUC-Amiga User's Club. Iscriviti e avrai a tua disposizione i migliori programmi. AUC – C.P. 244 – 07026 Olbia (SS) – Tel. 0789/22030 (Franco).

Alla Eporedia Computer Club puoi associarti senza spese e procurarti i migliori programmi per Amiga, C-128 e C-64. Scrivi a: E. C. C. – C. P. 14 – 10019 Strambino (TO) – Tel. 0125/712311 (ore serali).

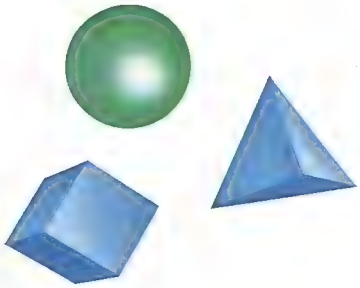
Amiga Computer Club cerca possessori di Amiga per scambio di informazioni e consigli. Se sei interessato, telefona a Fabio 059/561586. Solo zona Modena e prov.

Il Master Soft Club offre ai soci programmi di ogni genere, bollettino mensile, trucchi, novità, scambi, hardware vario ecc. Iscrizione gratuita e massima serietà assicurata. Nicola Gianno – Via Marsala, 351 – 91020 Rilievo (TP) – Tel. 0923/864559.

Il New Commodore Club visto il successo ottenuto con l'Amiga dal primo ottobre apre una sezione C-64 e una MS-DOS. Per informazioni scrivete a N.C.C. Via Baracca, 9/11 – 16035 Rapallo (GE), o telefonate allo 0185/60829.

INDICE DEGLI INSERZIONISTI

Inserzionista	Pag.
IHT Gruppo Editoriale	II, III, IV, 1, 8, 121
Informatica Italia	83
Lago	39
Sim Hi-Fi Ives	41



Direzione vendite spazi pubblicitari:

IHT Gruppo Editoriale
Commodore Gazette
Uffici Pubblicitari
Via Monte Napoleone, 9
20121 Milano
Tel. 02/794181 - 799492
792612 - 794122
Telex 334261 IHT I
Telefax 02/784021

Questo indice è da considerarsi come un servizio addizionale. L'Editore non si assume alcuna responsabilità per eventuali errori e/o omissioni.

Indirizzare eventuali lamentele riguardanti gli inserzionisti a:

Commodore Gazette
Uffici Pubblicitari
Relazioni Inserzionisti
Via Monte Napoleone, 9
20121 Milano

Nessuna responsabilità viene altresì assunta dalla Commodore Gazette per eventuali problemi di qualsiasi natura con gli inserzionisti. La responsabilità di quanto pubblicato negli spazi pubblicitari è esclusivamente del committente.

NEL PROSSIMO NUMERO

Luglio/Agosto 1989 (in edicola nel mese di settembre)

- **Dal nostro inviato negli Stati Uniti -** Le anteprime sull'Amiga direttamente dalla "Commodore Amiga Developers Conference 1989", svoltasi a San Francisco.
- **C-64/128 -** Interviste, recensioni, listati e curiosità per riscoprire il vostro C-64/128.
- **Prove stampanti -** Sotto i riflettori altre due stampanti: la Mannesmann MT81 e la Oki Microline 393.
- **Fare musica con l'Amiga -** Prima puntata di una serie dedicata a esperienze pratiche e recensioni di software e hardware musicale per l'Amiga.
- **Saloni USA -** Da Los Angeles tutte le novità presentate nel corso del World of Commodore Show.
- **Programmare l'Amiga -** Prova software del pacchetto TxEd Plus. Ancora sotto esame interrupt e temporizzazioni. Analisi dei file INCLUDE e della "nomenclatura" dell'Amiga.

SERVIZIO LETTORI

Questa scheda è valida fino al 15 settembre 1989

A. Come giudica questo numero di Commodore Gazette?

- ☐ 1. Ottimo
- ☐ 2. Molto buono
- ☐ 3. Buono
- ☐ 4. Discreto
- ☐ 5. Sufficiente
- ☐ 4. Mediocre
- ☐ 6. Insufficiente

B. Quale(i) articolo(i) di questo numero ha apprezzato maggiormente?_____

C. Quale(i) articolo(i) di questo numero giudica peggiore(i)?_____

D. Quali argomenti dovrebbero essere trattati nei prossimi numeri di Commodore Gazette?_____

E. Con quale aggettivo descriverebbe Commodore Gazette?_____

F. Quante persone leggono la sua copia di Commodore Gazette?

- ☐ 1. Uno
- ☐ 2. Due
- ☐ 3. Tre
- ☐ 4. Quattro o più

G. Ha dei suggerimenti?_____

H. Quale(i) computer utilizza?

- ☐ 1. C-64
- ☐ 2. C-128
- ☐ 3. C-128D
- ☐ 4. Amiga 500
- ☐ 5. Amiga 1000
- ☐ 6. Amiga 2000
- ☐ 7. PC 1
- ☐ 8. PC 10
- ☐ 9. PC 20
- ☐ 10. Altro (specificare)_____

I. Quale(i) computer intende acquistare nel futuro?

- ☐ 1. C-64
- ☐ 2. C-128D
- ☐ 3. Amiga 500
- ☐ 4. Amiga 2000
- ☐ 5. PC 1
- ☐ 6. PC 10
- ☐ 7. PC 20
- ☐ 8. Altro (specificare)_____

L. È un acquirente dei libri della IHT? Se sì come li giudica?_____

M. Quali altre riviste (sia d'informatica che non) legge abitualmente?_____

N. Indichi quali sono i suoi maggiori interessi

- ☐ 1. Videoregistrazione
- ☐ 2. Hi-Fi
- ☐ 3. Fotografia
- ☐ 4. Automobili
- ☐ 5. Sport
- ☐ 6. Viaggi

Nome _____
Cognome _____
Indirizzo _____
Città _____
Prov. _____ **C.a.p.** _____
Età _____ **Professione** _____

SCHEDA ORDINAZIONE LIBRI

Con il presente tagliando desidero ordinare i seguenti libri:

Collana Informatica	☐	L'Amiga (Michael Boom)	L. 60.000
	☐	Il Manuale dell'AmigaDOS (Commodore-Amiga)	L. 60.000
	☐	Programmare l'Amiga Vol. I (Eugene P. Mortimore)	uscita: dicembre '89
	☐	Programmare l'Amiga Vol. II (Eugene P. Mortimore)	uscita: settembre '89
	☐	Guida ufficiale alla programmazione di GEOS (Berkeley Softworks)	L. 64.000
	☐	Flight Simulator Co-pilot (Charles Gulick)	L. 30.000
	☐	Volare con Flight Simulator (Charles Gulick)	L. 45.000
Collana Cinema	☐	Le mille luci di Hollywood (David Chell)	L. 42.000
Collana Tempus	☐	Inventori del nostro tempo (Kenneth A. Brown)	L. 42.000
	☐	Computer in guerra: funzioneranno? (David Bellin & Gary Chapman)	L. 39.900

Pagherò in contrassegno al postino la somma di L.....+ spese postali (L. 6.000 per volume)

Nome e cognome _____
Indirizzo _____
Città _____
Prov. _____ **C.a.p.** _____ **Tel.** _____
Firma _____

☐ Desidero inserire gratuitamente un mio annuncio nella rubrica CLASSIFIED (solo per i privati e per gli annunci non a scopo di lucro).

☐ Desidero inserire un mio annuncio nella rubrica CLASSIFIED. Allego assegno bancario o circolare o postale oppure fotocopia della ricevuta del vaglia postale per un totale di L. _____ . Il mio codice fiscale o partita IVA (per le aziende) è il seguente: _____.

Attenzione: perché un annuncio venga accettato è necessario che sia stato compilato anche il questionario presente sull'altro lato di questo tagliando. Non si accettano fotocopie.

TESTO: _____

_____

_____

_____

Inserire all'interno di una busta affrancata e spedire a:

Commodore Gazette
Servizio Lettori
Via Monte Napoleone, 9
20121 Milano

Inserire all'interno di una busta affrancata e spedire a:

IHT Gruppo Editoriale
Divisione Libri
Via Monte Napoleone, 9
20121 Milano

UN COMPUTER, UN LIBRO...
...LE ALI

FLIGHT SIMULATOR CO-PILOT

Questo volume è un vero "istruttore di volo" per coloro che possiedono il programma Flight Simulator della Microsoft per i PC IBM e compatibili, oppure Flight Simulator II della SubLogic per Commodore 64, Apple II, Atari 800 XL e XE.

Non è un semplice compendio di comandi da ricordare a memoria, ma un brillante compagno d'avventura, scritto con estro e ironia. Un libro da sfogliare con il computer acceso, per imparare passo dopo passo i principi del volo "rettilineo e livellato", le procedure di decollo e di atterraggio, del volo strumentale e notturno, e del volo acrobatico.

152 pagine, L. 30.000

VOLARE CON FLIGHT SIMULATOR

Questo volume si rivolge a tutti gli utenti che possiedono la versione di Flight Simulator per l'Amiga, l'Atari ST o l'Apple Macintosh.

Con uno stile spigliato, ma con grande attenzione a ogni dettaglio, l'autore conduce l'allievo pilota attraverso le varie fasi di un corso di volo che si concluderà con un primo approccio al volo acrobatico. Il testo è completato da dettagliate cartine di rotta, schemi che illustrano l'uso dei comandi, parametri di volo ideali e procedure utili nelle più svariate situazioni di volo.

232 pagine, L. 45.000

I libri IHT sono disponibili nelle migliori librerie e computer shop. Per ordini diretti servirsi dell'apposito modulo pubblicato a pagina 127
IHT Gruppo Editoriale - Via Monte Napoleone, 9 - 20121 Milano - Tel. 02/794181-794122 - Fax 784021 - Telex 334261 IHT I
Distribuzione: Messaggerie Libri - Via Giulio Carcano, 32 - 20141 Milano - Tel. 02/8438141

TEMPUS™

NOVITÀ TEMPUS

Sono disponibili in libreria
i primi tre volumi della serie TEMPUS – Scienza e tecnologia, una collana che la IHT dedica alla storia, all'impatto sociale e al futuro della scienza e dell'alta tecnologia.

INVENTORI DEL NOSTRO TEMPO, di Kenneth A. Brown, raccoglie 16 interviste con alcuni dei più importanti inventori di questo secolo, gli ideatori del laser, del microprocessore, del pacemaker trapiantabile, del computer Apple II...

COMPUTER IN GUERRA: FUNZIONERANNO?, a cura di David Bellin e di Gary Chapman, è un volume scritto da scienziati e ricercatori che fanno capo all'associazione "Professionisti dell'informatica per la responsabilità sociale". Fornendo un'ampia documentazione, descrive i pericoli delle ricerche finanziate negli ultimi anni dal Pentagono (guerra automatizzata, armamenti autonomi, robot killer) e spiega perché un programma ambizioso come l'SDI non sarà mai realizzabile.

IL SEGRETO DEL SUCCESSO, di G. Ray Funkhouser e Robert R. Rothberg, analizza in ogni aspetto il fenomeno della crescita economica. Attraverso storie di clamorosi successi e di spettacolari fallimenti (come quello della Atari), spiega in quali casi la crescita aziendale è sana, e quando invece si risolverà probabilmente in un disastro.

IHT
GRUPPO EDITORIALE

I libri IHT sono disponibili nelle migliori librerie. Per ordini diretti servirsi del modulo pubblicato a pagina 127
IHT Gruppo Editoriale –Via Monte Napoleone, 9 – 20121 Milano – Tel. 02/794181-794122 – Fax 02/784021 – Telex 334261 IHT I
Distribuzione: Messaggerie Libri – Via Giulio Carcano, 32 – 20141 Milano – Tel. 02/8438141